Fayoum University

Faculty of Computers and Information

٢٠١٥

System Analysis of Public Opinion

Supervised By:

Dr: Haytham Al-Feel

Dean: Dr:

Nabila Mohammed Hassan

Team Members (A ٢H ٢SZ)

١. *Asmaa Soloma*
٢. *Asmaa Taha*
٣. *Sara Kamal*
٤. *Heba Hussein*
٥. *Hanaa Atwa*
٦. *Zahraa Ahmed*

TABLE OF CONTENTS

Acknowledgment---------- ١

Abstract ---------- ٢

Chapter 1: Introduction

١.١ Overview---------- ٤

١.٢ Problem Definition---------- ٥

١.٣ Objectives---------- ٦

١.٤ Documentation Structure---------- ٧

Chapter 2: Background

٢.١ Social Network----------٨

٢.٢ Role of Social Network in Egyptian Revolution ----------٨

٢.٣ Examples of Social Network----------١٢

٢.٤ Social Network with Our System ----------١٦

٢.٥ Input----------١٧

٢.٦ Advantages of System----------١٧

٢.٧ Conclusion----------١٨

Chapter 3: Related Work

٣.١ Overview----------١٩

٣.٢ Research and Design of Internet Public Opinion Analysis System----------١٩

٣.٣ Eurobarometer Interactive Search System----------٢٠

٣.٤ The Design of Public Opinion Analysis System Based On Topic Events----------٢٠

٣.٥ Network Opinion Analysis System----------٢٢

٣.٦ Differences between Our System and Others----------٢٢

٣.٧ Conclusion---------- --٢٣

Chapter 4: System Development Life Cycle

٤.١ SDLC----------٢٤

٤.١.١ Overview----------٢٥

٤.١.٢ System Development Phase----------٢٦

٤.٢ Planning ----------٢٩

٤.٢.١ Overview ----------٢٩

٤.٢.٢ Planning Steps---------- ٢٩

٤.٢.٣ The Milestone for Planning Phase----------٣٠

٤.٣ System Analysis----------٣١

٤.٣.١System's Characteristics ----------٣٢

٤.٣.١.١ Purpose----------٣٢

٤.٣.١.٢ Input----------٣٢

٤.٣.١.٣ Output----------٣٢

٤.٣.٢ Constrains----------٣٣

٤.٣.٣ Components----------٣٣

٤.٣.٤ Environment----------٣٣

٤.٣.٥ Interrelationship----------٣٣

٤.٣.٦ System Boundary----------٣٣

٤.٣.٧ Interface----------٣٤

٤.٣.٨ System Requirements----------٣٤

٤.٣.٨.١ Functional Requirements----------٣٤

٤.٣.٨.٢ Nonfunctional Requirements----------٣٤

٤.٤ System Design ----------٣٧

٤.٤.١ Analysis Use Case----------٣٧

٤.٤.٢ Sequence Diagram----------٤٣

٤.٤.٣ Collaboration Diagram----------٤٨

٤.٤.٤ Activity Diagram----------٥٢

٤.٤.٥ Class Diagram----------٥٧

٤.٥ System Implementation----------٥٨

٤.٦ Maintenance----------٥٩

Chapter 5: Data Mining

٥.١ Overview----------٦٢

٥.٢ Data, Information and Knowledge----------٦٢

٥.٢.١ Data----------٦٣

٥.٢.٢ Information----------٦٣

٥.٢.٣ Knowledge----------٦٣

٥.٣ What can data mining do? ----------٦٤

٥.٤ How does data mining work? ----------٦٤

٥.٥ Stemming----------٦٥

٥.٦ Data Mining in "System Analysis of Public Opinion"---٦٦

٥.٧ Twitter Data Collector Tool----------١٤٠

٥.٨ Conclusion----------١٤٥

Chapter 6: Semantic Web

٦.١ Introduction----------١٤٦

٦.٢ The Development Methodology----------١٤٧

٦.٣ Installation Tool Techniques----------١٤٩

٦.٣.١ Install Protégé----------١٤٩

٦.٣.٢ Creating Ontology ----------١٥٥

٦.٣.٣ Install Jess and Building SWRL----------١٦٦

٦.٣.٤ Install and Run Jena----------١٧٣

٦.٤ SPARQL Query---------- ------١٨٦

Chapter 7: Case Study

٧.١ Introduction----------١٩٢

٧.٢ Final Result----------١٩٢

Chapter 8: Conclusion & Future Work

٨.١ Introduction----------١٩٥

٨.٢ Difficulties That Faced Us----------١٩٥

٨.٣ Future Work----------١٩٥

٨.٤ Conclusion----------١٩٦

References----------١٩٧

Glossary----------١٩٩

AppendixA (Tweets Preparation) ----------٢٠٣

ملخص----------٢٠٩

LIST OF FIGURES

Figure Number	Description	Page Number
١	Chapter ٢ :Figure ٢.٣ Twitter Home Page	١٣
٢	Chapter ٢ : Figure ٢.٣ Twitter Logo	١٤
٣	Chapter ٣ : Figure ٣.٣ Generate Report for Public Opinion	٢٠
٤	Chapter ٣ : figure ٣.٣ Generate Report of Public Opinion	٢١
٥	Chapter ٤ : Figure ٤.٢ SDLC	٢٤
٦	Chapter ٤ : Figure ٤.٤.١ Use Case Diagram	٣٧
٧	Chapter ٤ : Figure ٤.٤.٢ Sequence Diagram of Login	٤٣
٨	Chapter ٤ : Figure ٤.٤.٢ Sequence Diagram of Adding Administrator	٤٤
٩	Chapter ٤ : Figure ٤.٤.٢ Sequence Diagram of Removing Administrator	٤٥
١٠	Chapter ٤ : Figure ٤.٤.٢ Sequence Diagram of Giving Permission	٤٦
١١	Chapter ٤ : figure ٤.٤.٢ Sequence Diagram of Generating Report	٤٧

Figure Number	Description	Page Number
١٢	Chapter ٤ : Figure ٤.٤.٣ Collaboration Diagram for Adding Administrator	٤٨
١٣	Chapter ٤ : Figure ٤.٤.٣ Collaboration Diagram for Removing Administrator	٤٩
١٤	Chapter ٤ : Figure ٤.٤.٣ Collaboration Diagram for Giving Permission	٥٠
١٥	Chapter ٤ : Figure ٤.٤.٣ Collaboration Diagram for Generating Report	٥١
١٦	Figure ٤.٤.٤ Activity for Login	٥٢
١٧	Figure ٤.٤.٤ Activity for Adding Administrator	٥٣
١٨	Figure ٤.٤.٤ Activity for Removing Administrator	٥٤
١٩	Figure ٤.٤.٤ Activity for Giving Permission	٥٥
٢٠	Figure ٤.٤.٤ Activity for Generating Report	٥٦
٢١	Figure ٤.٤.٥ Class Diagram	٥٧
٢٢	Figure ٥.٦(١) Result of Data Mining	١٣٨
٢٣	Figure ٥.٦(٢) Result of Data Mining	١٣٨

Figure Number	Description	Page Number
٢٤	Figure ٥.٦(٣) Result of Data Mining	١٣٩
٢٥	Figure ٥.٦(٤) Result of Data Mining	١٣٩
٢٦	Figure ٥.٦(٥) Result of Data Mining	١٤٠
٢٧	Figure ٥.٧ Twitter Data Collector	١٤٠
٢٨	Figure ٥.٧ Encrypted Data Of Tool	١٤١
٢٩	Figure ٥.٧ Result After Encoding	١٤٤
٣٠	Figure ٦.١ Semantic Web	١٤٧
٣١	Figure ٦.٣.١ Install Protégé (١)	١٤٩
٣٢	Figure ٦.٣.١ Install Protégé (٢)	١٥٠
٣٣	Figure ٦.٣.١ Install Protégé (٣)	١٥١
٣٤	Figure ٦.٣.١ Install Protégé (٤)	١٥٢
٣٥	Figure ٦.٣.١ Install Protégé (٥)	١٥٣
٣٦	Figure ٦.٣.١ Install Protégé (٦)	١٥٤
٣٧	Figure ٦.٣.٢ Create Ontology(١)	١٥٥
٣٨	Figure ٦.٣.٢ Create Ontology(٢)	١٥٦

Figure Number	Description	Page Number
٣٩	Figure ٦.٣.٢ Create Ontology(٣)	١٥٧
٤٠	Figure ٦.٣.٢ Create Ontology(٤)	١٥٨
٤١	Figure ٦.٣.٢ Create Ontology(٥)	١٥٩
٤٢	Figure ٦.٣.٢ Create Ontology(٦)	١٦٠
٤٣	Figure ٦.٣.٢ Create Ontology(٧)	١٦١
٤٤	Figure ٦.٣.٢ Create Ontology(٨)	١٦٢
٤٥	Figure ٦.٣.٢ Create Ontology(٩)	١٦٣
٤٦	Figure ٦.٣.٢ RDF Graph	١٦٥
٤٧	Figure ٦.٣.٣ Jess Preferences	١٦٧
٤٨	Figure ٦.٣.٣ SWRL Rule	١٧١
٤٩	Figure ٦.٣.٤ Create Java Class	١٧٥
٥٠	Figure ٦.٣.٤ Create Java Class	١٧٦
٥١	Figure ٦.٣.٤ Add Library	١٧٩
٥٢	Figure ٦.٤.٣ Add Library	١٨٠
٥٣	Figure ٦.٤.٣ Select Jar File	١٨٢
٥٤	Figure ٦.٤.٣ Select Jar File	١٨٣

Figure Number	Description	Page Number
٥٥	Figure ٦.٤.٣ Add Library	١٨٤
٥٦	Figure ٦.٤.٣ Add Library	١٨٥
٥٧	Figure ٦.٤.٣ Add RDFA	١٨٩
٥٨	Figure ٦.٤.٣ Add RDFA	١٩٠
٥٩	Figure ٧.٢ Login	١٩٢
٦٠	Figure ٧.٢ Add People to System	١٩٣
٦١	Figure ٧.٢ Final Result	١٩٤

List of Tables

Table Number	Description	Page Number
١	Chapter ٤ : Login Use Case Analysis	٣٨
٢	Chapter ٤ : Adding New Person Use Case Analysis	٣٩
٣	Chapter ٤ : Removing Person Use Case Analysis	٤٠
٤	Chapter ٤ : Giving Permission Use Case Analysis	٤١
٥	Chapter ٤ : Generating Report Use Case Analysis	٤٢

List of ABBREVIATIONS

ABBREVIATION	Stands for
SVM	Support Vector Machine
SDLC	System Development Life Cycle
XP	Extreme Programming
PLC	Project Life Cycle
RDF	Recourse Description Framework
GUI	Graphical User Interface
HTML	Hyper Text Markup Language
XHTML	ExtensibleHyper Text Markup Language
XML	Extensible Markup Language
OWL	Web Ontology Language
W٣C	World Wide Web Consortium
SWRL	Semantic Web Rule Language

ACKNOWLEDGMENT

The success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

The special thank goes to our helpful supervisor, **Dr.Hythem El-feel**. The supervision and support he gave truly help in the progression of our graduation project. The co-operation is much indeed appreciated.

Our greatly appreciate to the contribution of our faculty, Faculty of Computer and Information (FCI). Especially **Prof.Nabila Hassan**, listen to our problems and did her best to satisfy all groups.

At Last but not least we'd like to thank **Dr.RashaBadry** for her big effort with us in this project.

ABSTRACT

Many of the revolutions that have affected the current political situation led to the occurrence of multiple political views among most people, which led to the emergence of a clear contrast between those views.

"System Analysis for Public Opinions (SAPO)"is software application that tries to classify political views among supporters and opponents through their opinion from social media specially twitter.

SAPO provide public view about specific political situation what happened and what will happen and what is expected to happen and contributes to improve the political vision future.

SAPO it is about a web application that take data from social media specially twitter through some tools making mining for this data and using semantic web and ontology which lead to classify the opinion for against the current in our country.

Objectives of Our Project SAPO

- Classify public opinion into for or against.
- Determine opinion that belongs to political state or related to politics.
- Determine if people opinion is creative or not, if it is not creative determine the correctness for this opinion.
- Remove the mask from people who attack our country.

The Expected Results

Determine people who supporters and opponents what happened and what will happen and what are expected to happen and contributes to improve the political vision future.

Chapter 1

ction

١.١ Overview
١.٢ Problem Definition
١.٣ Objectives
١.٤ Documentation Structure

١.١ Overview

In the last few years there were many problems that face our country due to misunderstanding of what people need, what is the wrong with this situation? What must we do?

For example "Egyptian Revolution in ٢٠١١", The Egyptian protesters" target was the main powers of the tyrannical regime and President Hosni Mubarak viewed as the main principal of the authoritarian regime. This time, the leading of objection was not human beings who can be imprisoned and repressed by the regime.

It was a new source, born from technology that had been utilized by intelligent minds over the past three decades Social networks played a significant role in the North African political revolutions.

Mubarak guaranteed that he would not attend the next election and appointed Omar Suleimanas his Vice-President, but none of protesters were sufficiently satisfied to leave the Tahrir Square of Cairo and continued to their civil disobediences.

The threat of being fired upon and possible death of protesters was not enough to disperse the crowd from Tahrir Square and other streets. The same scenario happened in other cities of Egypt. The march of the million was underway. Civil disobedience was the turning point of rebellion. Civil resistance was efficacious and Mubarak's regime collapsed.

Unemployment, inflation, repression and state corruption were the most important causes of Egypt's uprising, as the need to comprehensive reform was viewed an important step toward a great future.

From our project we try to be in the heart of the event, try to understand thoughts of youth, predict what will happen, determine the problem that face people in the current time from these we can find many solutions that will prevent people' anger.

١.٢ Problem Definition

Due to the political changes and many of revolutions that affects the progress of our country on all fields, such as social, economy……and other fields.

People are divided into two parts, one is for and one is against, we try to analyze the opinion view to determine who is for and who is against, we want to know if this Opposition is constructive or destructive to reveal who plots against our land home.

From this project we can determine problem that face people in all fields not only political, this will help us to cope with all changes of all fields and reveal its effects in people.

System's Actor

Youth is the backbone of any country; from our project we try to focus on youth who is the backbone of any country and they play the main part of any change.

Youth from ١٥ to ٣٠ the most user of social network and modern technology who we must focus on their views and know how they think, what they need, they arethe mirrors of our country.

From analyzing their effective participation over the social network such as facebook, twitter and other kinds of social network we can determine the state of our country.

After this we try to find solution that will cover the problem without any revolution, step by step we provide appropriate solutions that will satisfy this situation.

١.٣ Objectives

From all events that happen in our country and many changes of political decisions that affect on all fields that lead to many revolutions we decide to contribute in offering solution to these problems.

There are many things that motivate us to make "System analysis of public opinion".

- Analysis of public opinion to determine the current state of our country.
- How to reveal who plots against our land home.
- To determine who is for or against the current state.
- To know if this Opposition is constructive or destructive to reveal who plots against our land home.
- To contribute in the progress of our land home from this system.

١.٤ Documentation Structure

- **Chapter ١**

  This chapter is an overview about our project including objectives of this project and its role in our county's progress.

- **Chapter ٢**

  This chapter is a background about our project; in this chapter we will discuss Social Network and its types.

  Social Network plays an important role in the Egyptian revolution.

- **Chapter ٣**

  This chapter includes related work in this field of analysis of public opinion.

- **Chapter ٤**

  This chapter includes System Development Life Cycle Applying on this system.

- **Chapter ٥**

  This chapter includes Data Mining and its role in this system.

- **Chapter ٦**

  This chapter includes Semantic web and its role in the system.

❖ **Chapter ٧**

This Chapter includes final results of our system.

❖ **Chapter ٨**

This chapter includes difficulties that faced usand future work.

Chapter 2
Background

٢.١ Social Network

٢.٢ Role of Social Network in Revolution

٢.٣ Examples of Social Network

٢.٤ Social Network with Our System

٢.٥Input

٢.٦ Advantage of System

٢.٧ Conclusion

٢.١ Social Network

Alternatively referredto as a virtual community or profilesite**,** a social networkis a website on the Internet that brings people together in a central location to talk, share ideas and interests, or make new friends. This type of collaboration and sharing of data is often referred to as social media. Unlike traditional media that is often created by no more than ١٠ people, social media sites contain content that has been created by hundreds or even millions of different people. Below is a small list of some of the biggest social networks used today.

Social networks like Facebook, MySpace and LinkedIn are great ways of keeping in touch with friends and family around the world as well as making new connections with people based on similar interests or professions. There are tons of deferent social networks that you can join – all for free.

٢.٢ Role of Social Network in Egyptian Revolution

What happened in January ٢٠١١ in Egypt did not start in January ٢٠١١. It began at least ten years earlier, and it's not over yet. The revolution was joined by people of all walks of life, Internet users and non-users alike.

It gained momentum once it was joined by hundreds of thousands of workers, many of whom have been demonstrating for years. Why was this particular round of demonstrations so successful? Much of the organization and mobilization occurred through the Internet, particularly on social media such as Facebook and Twitter. But social media also played a vital role as a democratic model.

Its inclusive space indirectly taught lessons in democracy to a wide sector of Egyptian youth that was not necessarily politically inclined. When the right moment arrived, they were ready to join the revolt.

The main catalyst for the January ٢٥ revolution was the Internet, so it may be accurate to describe this as an Internet-based revolution. Not that the Internet was the only factor involved, or that Internet users were the only ones protesting. But the Internet was the tool that showed every dissident voice in Egypt that he or she is not alone, and is indeed joined by at least hundreds of thousands who seek change.

Facebook did not go to Tahrir Square. The people did, Twitter did not go to Al-Qaied Ibrahim Square. The people did. More than one-third of Egypt's population of eighty million remains illiterate, and just ٢٥ percent of Egyptians use the Internet. However, Facebook and Twitter were instrumental in organizing, motivating, and directing these crowds as to where to go and what to do.

Egypt's revolution was created as an event on Facebook eleven days in advance. People clicked "I'm attending." Certainly, this was a people's revolution, yet one based on and accelerated in many ways by the Internet. What happened in Tahrir and every square in Egypt was the accumulation of years and years of activism, including Internet activism. Social media prepared Egyptians for the revolution and enabled them to capitalize on an opportunity for change when the time came.

The Internet, by definition, is a democratic medium, at least in the sense that anyone with Internet access is a potential publisher of information. The average person may not have a chance to publish a newspaper article, or even a letter to the editor, and may not have a chance to appear on television, or to call in to a program.

But they can readily design a website, publish a blog, or have a page on the numerous social networking sites, whereby they can make their views public. The mere presence of the Internet as a source of information helps open up a freer space for public debate, and makes it much more difficult for governments to censor information.

When regimes censor an article in a magazine or an entire edition of a newspaper, that article or newspaper will find its way on to the Internet and in people's email boxes.

Internet activism started in Egypt with the appearance of Web ٢.٠ technology in the country around ٢٠٠٣. The new generation of interactive applications that took over the Internet since then has enabled and empowered the Internet user to do with the tools what was never possible before.

This started with very simple tools, such as enabling readers to leave comments on a news story, and soon proliferated to include applications that changed the face of the Internet through making it much easier to have user-generated content. Blogging was the first valuable brainchild of Web ٢.٠ technologies.

A blog, short for 'Web log,' is an Internet personalized space where someone can 'blog' or write their own thoughts about anything they please. The Internet has always been a user-based platform.

But before Web ٢.٠, the options were very limited. One had to learn HTML (Web design language) to be able to build a website. The on-line services that enabled a user to create a 'page' from some templates were very limited both in size and in design options. Blogging changed that forever.

٢.٣ Examples of Social Network

- **Facebook**

By far the fastest growing social network, Facebook has grown from just a college-age group network to a global, multi-age, multi-ethnic site.

Hundreds of millions of people are on Facebook and millions more join every month! On Facebook you can connect with friends and family by searching for names or by view friends of other people you're already connected to, as well as through on site 'networks' that are organized around school, location or workplace.

- **MySpace**

MySpace was one of the first really big social networks, and while it still has over ٢٠٠ million users, its use has been steadily declining as Facebook grows.

MySpace tends to attract a younger demographic than Facebook, and is particular popular with those involved in the music scene given the site's fantastic ability to promote bands and other music artists exceedingly well.

- **LinkedIn**

LinkedIn is a professional social network designed for business people or working professionals to make connections with colleagues and other business contacts.

It can also serve as a means to make contact with potential customers, clients and partners. It allows members to post resume and work history as well as get recommendations from colleagues and former employers and employees.

- **Friendster**

A popular social network that brings together friends, family, and allows you to meet new people who share similar interests to you from all over the world.

- **Pinterest**

An upcoming and popular picture and sharing service that allows anyone to easily share pictures, create collections, and more.

- **Twitter**

Twitter is an online social networking service that enables users to send and read short ١٤٠-character messages called "tweets".

Registered users can read and post tweets, but unregistered users can only read them. Users access Twitter through the website interface, SMS, or mobile device app. Twitter Inc. is based in San Francisco and has more than ٢٥ offices around the world.

Twitter was created in March ٢٠٠٦ by Jack Dorsey, Evan Williams, Biz Stone and Noah Glass and launched by July ٢٠٠٦. The service rapidly gained worldwide popularity, with more than ١٠٠ million users who in ٢٠١٢ posted ٣٤٠ million tweets per day. The service also handled ١.٦ billion search queries per day. In ٢٠١٣ Twitter was one of the ten most-visited websites, and has been described as "the SMS of the Internet." As of May ٢٠١٥, Twitter has more than ٥٠٠ million users, out of which more than٣٠٢ million are active users.

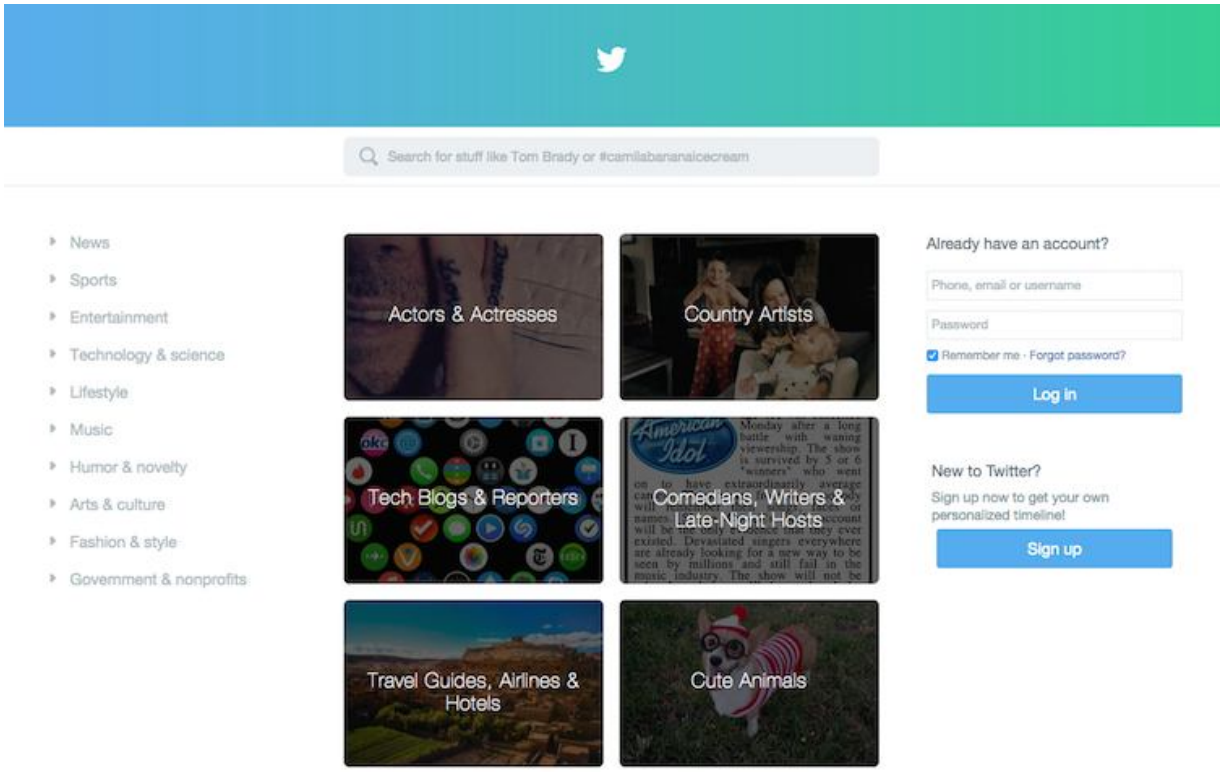


Figure ٢.٣ twitter home page

Logo

Twitter has become internationally identifiable by its signature bird logo. The original logo was in use from its launch in March ٢٠٠٦ until September ٢٠١٠. A slightly modified version succeeded the first style when the website underwent its first redesign.

Figure ٢.٣ twitter logo

Format of Tweet

Users can group posts together by topic or type by use of hashtags – words or phrases prefixed with a "#" sign. Similarly, the "@" sign followed by a username is used for mentioning or replying to other users. To repost a message from another Twitter user and share it with one's own followers, a user can click the retweet button within the Tweet.

Twitter Revolution in Egypt

In Egypt Revolution of ٢٠١١, the oppositional movement against the ruling of Mubarak was active on various platforms of social media. For example, “the hashtag #Jan٢٥th was used to mobilize protesters on Twitter” to join the demonstration on Jan ٢٥th on Tahrir Square. Along with other methods such as text message, flyers and words of mouth, it drew a crowd of ٨٠,٠٠٠ to the street of Cairo on that day. Similar to its Iranian correspondent, Egypt government shut down the access to Twitter in the afternoon on the day of gathering. The connection was not restored until February ٢.

Moreover, Twitter was applied to communicate with the audience outside Egypt to "globalized the movement and win international support to protect and sustain the uprising". The worldwide audience was also able to have constant update with the situation in Egypt, besides simply listening to the State's point of view. As consequences, the revolution succeeded in the resignation of Mubarak in February ١١, ending his dictatorship lasted for over ٣ decades. An article in the magazine Wired states that social media did not cause the Egypt revolution. Rather, Twitter and Facebook were more like "a spark and an accelerant", "catalyzing pro-democracy movements". They have had the most potent impact in "what has shocked most observers of the current Egyptian scene: the sheer speed with which the regime fell — ١٨ days".

٢.٤ Social Network with Our System

System analysis of public opinion is a web site that works over the social network.

From social media "System analysis of public opinion" works by making analysis of people's opinion, people are divided into two parts, one is for and one is against, we try to analyze the opinion view to determine who is for and who is against, we want to know if this Opposition is constructive or destructive to reveal who plots against our land home.

From this social network we can determine problem that face people in all fields not only political, this will help us to cope with all changes of all fields and reveal its effects in people.

Youth is the backbone of any country; from our project we try to focus on youth who is the backbone of any country and they play the main part of any change.

Youth from ١٥ to ٣٠ the most user of social network and modern technology who we must focus on their views and know how they think, what they need, they arethe mirrors of our country.

From analyzing their effective participation over the social network such as facebook, twitter and other kinds of social network we can determine the state of our country.

After this we try to find solution that will cover the problem without any revolution, step by step we provide appropriate solutions that will satisfy this situation.

٢.٥ Input

System's inputs are all things that are published over social network such as Facebook, LinkedIn and Twitter.

In our system the inputs are hash tags that describe each subject, from these tweets our system works by analyzing it and extract the problem that face people in the current state.

٢.٦ Advantagesof System

There many advantages that comes from this system such as

- Analysis of public opinion to determine the current state of our country.
- Revealing who plots against our land home.
- Determining who is for or against the current state.
- Knowing if this Opposition is constructive or destructive to reveal who plots against our land home.
- Contributing in the progress of our land home from this system.
- Solving problems early.

٢.٧ Conclusion

This chapter includes Social Network and its role in Egyptian Revolution and examples of social networks such as Facebook, Twitter, and LinkedIn and so on.

System Analysis of Public Opinion is a web site that deal with social network special with Twitter until now to analysis what people share among them and classify them into Supporter and opposition for the current situation.

Chapter 3

l Work

٣.١ Overview

٣.٢ Research Design of Internet Public Opinion Analysis System

٣.٣ Eurobarometer Interactive Search System

٣.٤ The Design of Public Opinion Based on Topic Events

٣.٥ Network Opinion Analysis System

٣.٦ Difference between Our System and Others

٣.٧ Conclusion

٣.١ Overview

There are other systems that make analysis of public opinion, but they differ from our system as they do analysis for public opinion in specific topics such as:

١. Life Satisfaction
٢. Satisfaction with national and EU democracy
٣. Unification
٤. Membership to the EU
٥. If EU scrapped

These systems give you options to enter specific topic or to choose from list of topics.

٣.٢ Research and Design of Internet Public Opinion Analysis System

Internet is becoming a spreading platform for the public opinion. It is important to grasp the Internet public opinion in time and understand the trends of their opinion correctly. Text classification plays a fundamental role in a number of information management and retrieval tasks. But Web-page classification is much more difficult than pure-text classification due to a large variety of noisy information embedded in Web pages**.**

By applying Web-page classification through summarization to extract the most relevant content from the Web pages and then pass them to standard text classification algorithms (NB or SVM).by using text classification and text clustering algorithms, which have been shown to be efficient and effective for singly using.

٣.٣ Eurobarometer Interactive Search System

This tool runs a search of the Trends questions database. It is possible to look for data on specific countries or a group of countries. In addition to comparing the results of several countries for a specific question, it allows analysis of country trends. The data can be displayed as a graph, a pie chart, in an excel table or in columns.

Survey results can also be displayed on a map of the EU. The color coding facilitates quick and easy comparisons between countries.

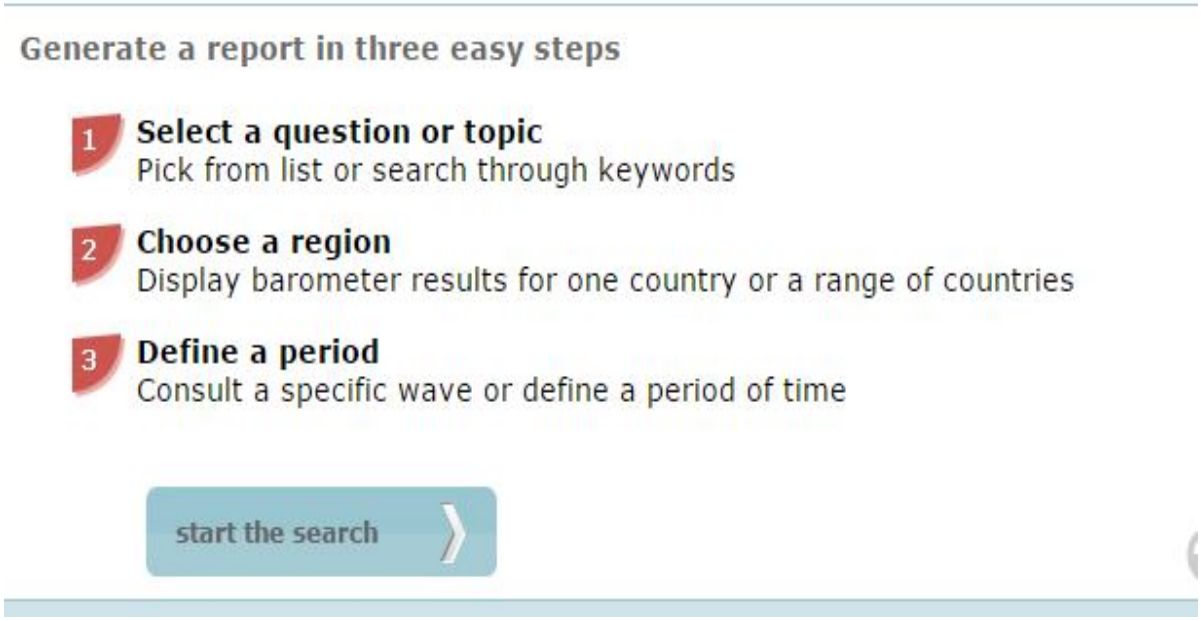


Figure ٣.٣ three steps to generate report for public opinion

Figure ٣.٣ step by step to generate report of public opinion

٣.٤ The Design of Public Opinion Analysis System Based on Topic Events

Public opinion analysis of the current system are generally aimed at one industry or the whole Internet, which led to excessive depth of lack of depth, to the subject of an Internet event information for in-depth analysis of public opinion.

Through this can public opinion to a specific event information for in-depth analysis of the multi-dimensional, the use of such public opinion analysis system for Government or large companies found that people focus on specific subject matter, social relations and events that exist within some hidden, Realization of the principle of association rules based on the analysis of social relations, and analyzed by both the support and confidence of the community property of the relevant entities, the module can be more accurately identify potential events related to the page through social entity-relationship of the mining algorithm social relations and can be used as recommendations for Government or corporate decisions, related technical research in the field to have some reference value.

٣.٥ Network Opinion Analysis System

Network Opinion Analysis System is based on information discovery and intelligent analysis. It uses the data mining technology and natural language processing technology , combining with the text information processing technology, natural language processing technology and the latest research result of artificial intelligence, information retrieval, text mining.

The development of application can be easily committed on it, and realizing the application for intelligent mining and knowledge management. Network opinion analysis system is developed based on our profound understanding for customer requirement and rich accumulation in technology.

٣.٦ Differences between Our System and the Others

The pervious systems work on specific topics compared result of one region with others.

But in System Analysis of Public Opinion work on social networks. It analysis that people share between them. Until now it works on political field and it can works on other fields.

System Analysis of public Opinion draws percentage of support and Oppositionof current situation.

٣.٧ Conclusion

This chapter about related work in analysis of public opinion and what is the difference between System Analysis of Public Opinion and other systems?
It shows how other systems work in this field.

Chapter 4

System Development Life Cycle

٤.١ SDLC
٤.٢ Planning
٤.٣ System Analysis
٤.٤ System Design
٤.٥ System Implementation
٤.٦ Maintenance

The systems development life cycle (SDLC), also referred to as the application development life-cycle, is a term used in systems engineering, information systems and software engineering to describe a process for planning, creating, testing, and deploying an information system.

The systems development life-cycle concept applies to a range of hardware and software configurations, as a system can be composed of hardware only, software only, or a combination of both.

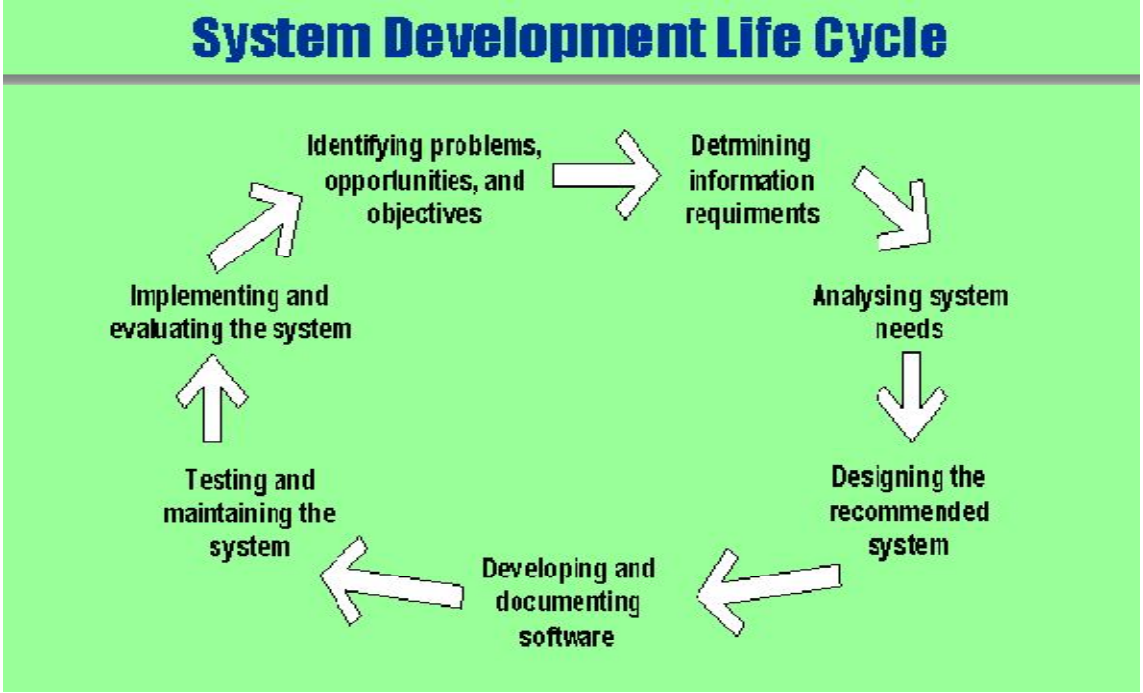


Figure ٤.١ SDLC

٤.١.١ Overview

A systems development life cycle is composed of a number of clearly defined and distinct work phases which are used by systems engineers and systems developers to plan for, design, build, test, and deliver information systems.

Like anything that is manufactured on an assembly line, an SDLC aims to produce high quality systems that meet or exceed customer expectations, based on customer requirements, by delivering systems which move through each clearly defined phase, within scheduled time-frames and cost estimates.

Computer systems are complex and often (especially with the recent rise of service-oriented architecture) link multiple traditional systems potentially supplied by different software vendors. To manage this level of complexity, a number of SDLC models or methodologies have been created, such as "waterfall"; "spiral"; "Agile software development"; "rapid prototyping"; "incremental"; and "synchronize and stabilize".

SDLC can be described along a spectrum of agile to iterative to sequential. Agile methodologies, such as XP and Scrum, focus on lightweight processes which allow for rapid changes (without necessarily following the pattern of SDLC approach) along the development cycle.

Iterative methodologies, such as Rational Unified Process and dynamic systems development method, focus on limited project scope and expanding or improving products by multiple iterations. Sequential or big-design-up-front (BDUF) models, such as waterfall, focus on complete and correct planning to guide large projects and risks to successful and predictable results.

Other models, such as anamorphic development, tend to focus on a form of development that is guided by project scope and adaptive iterations of feature development.

In project management a project can be defined both with a project life cycle (PLC) and an SDLC during which slightly different activities

occur. According to Taylor (٢٠٠٤) "the project life cycle encompasses all the activities of the project, while the systems development life cycle focuses on realizing the product requirements".

SDLC is used during the development of an IT project; it describes the different stages involved in the project from the drawing board, through the completion of the project.

٤.١.٢ System Development Phases

١. Planning
٢. Requirements Analysis
٣. Design
٤. Implementation
٥. Testing
٦. Evaluation

١. System Planning

The Planning phase is the most crucial step in creating a successful system, during this phase you decide exactly what you want to do and the problems you're trying to solve, by:

- Defining the problems, the objectives and the resources such as personnel and costs.
- Studying the ability of proposing alternative solutions after meeting with clients, suppliers, consultants and employees.
- Studying how to make your product better than your competitors.

٢. System Analysis

The end-user's requirements should be determined and documented, what their expectations are for the system, and how it will perform. A feasibility study will be made for the project as well, involving determining whether it's organizationally, economically, socially, technologically feasible.

It's very important to maintain strong communication level with the clients to make sure you have a clear vision of the finished product and its function.

٣.System Design

The design phase comes after a good understanding of customer's requirements; this phase defines the elements of a system, the components, the security level, modules, architecture and the different interfaces and type of data that goes through the system.

A general system design can be done with a pen and a piece of paper to determine how the system will look like and how it will function, and then a detailed and expanded system design is produced, and it will meet all functional and technical requirements, logically and physically.

٤.Implementation and Deployment

This phase comes after a complete understanding of system requirements and specifications; it's the actual construction process after having a complete and illustrated design for the requested system.

In the Software Development Life Cycle, the actual code is written here, and if the system contains hardware, then the implementation phase will contain configuration and fine-tuning for the hardware to meet certain requirements and functions.

In this phase, the system is ready to be deployed and installed in customer's premises, ready to become running, live and productive, training may be required for end users to make sure they know how to use the system and to get familiar with it, the implementation phase may take a long time and that depends on the complexity of the system and the solution it presents.

٥.System Testing and Integration

Bringing different components and subsystems together to create the whole integrated system, and then introducing the system to different inputs to obtain and analyze its outputs and behavior and the way it functions. Testing is becoming more and more important to ensure customer's satisfaction, and it requires no knowledge in coding, hardware configuration or design.

Testing can be performed by real users, or by a team of specialized personnel, it can also be systematic and automated to ensure that the actual outcomes are compared and equal to the predicted and desired outcomes.

٦. System Maintenance

In this phase, periodic maintenance for the system will be carried out to make sure that the system won't become obsolete, this will include replacing the old hardware and continuously evaluating system's performance, it also includes providing latest updates for certain components to make sure it meets the right standards and the latest technologies to face current security threats.

٤.٢ Planning

٤.٢.١ Overview

Planning is a process for accomplishing process. It is a blue of business growth and a road map of development. it helps in deciding objectives both in quantitative and qualitative terms. It is setting of goals on the basis of objectives and keeping in the resources.

What should a plan be?

A plan should be a realistic view of the expectations. Depending upon the activities a plan can be long range, intermediate range or short range. It is the framework within which it must operate. For management seeking external support, the plan is the most important document and key to growth. Preparation of a comprehensive plan will not guarantee success, but lack of a sound plan will almost certainly ensure failure.

٤.٢.٢ Planning Steps

Planning can be summarized in ٣ easy steps:

- Choosing a destination
- Evaluating alternative routes
- Deciding the specific course of your plan.

Helps management to clarify, focus, and research their businesses or project's development and prospects.

- Provides a considered and logical framework within which a business can develop and pursue business strategies over the next three to five years.
- Offers a benchmark against which actual performance can be measured and reviewed.

The term is also used for describing the formal procedures used in such an endeavor, such as the creation of documents, diagrams, or meetings to discuss the important issues to be addressed, the objectives to be met, and the strategy to be followed.

Beyond this, planning has a different meaning depending on the political or economic context in which it is used.

Two attitudes to planning need to be held in tension:

- On the one hand to be prepared for what may lie ahead, this may mean contingencies and flexible processes.
- On the other hand, our future is shaped by consequences of our own planning and actions.

This initial phase starts by defining the need. The objective may vary a great deal in nature and form.

In the planning phase the team needs to thoroughly understand the business model of the customer and their current IT state. They document the customer's inventorying application, information, and technology resources, record their locations, and analyze and prioritize their usefulness to business process.

After considering many business and IT factors, the team documents a plan, which identifies and prioritizes projects that will move the organization closer to the desired architecture.

٤.٢.٣ The Milestone for Planning Phase

The milestone for the planning phase is the project plan Approved milestone, in which the customer approves of the plan authorizes the next step.

During the release planning phase, feasibility studies are performed, user requirements are defined, high level estimates are produced and compared to forecast of the performing organization's capacity, assist in project selection.

Scenarios may be created to investigate possible combinations of projects (features) that would be affordable based on the capacity of the performing organization. Many of these activities are performed on an ongoing basis as new customer requests are submitted.

The Release Planning Review Board is responsible for evaluating, prioritizing and determining which service requests will be include in the next release. They ultimately direct the fate of the release and are the key decision-makers in the relative prioritization of requests and resource commitments.

The Release Planning Review Board is appointed by the OCE and should include the project sponsor and representative(s) (Director/V.P.level) from Product Management.

With the exception of the Project Sponsor and program manager assigned to release, the Release Planning Review Board member remains constant release after release. The Release Planning Review Board may or may not be the same the Gate Review Board defined below depending on how the organization structure is defined.

The planning stage of the system development life cycle begins with a request to the system analyst; this is known as the system request.

Our Plan

- Analysis and design(data mining) from ١/١٢/٢٠١٤ to ١/١/٢٠١٥
- Implementation from ١/٢/٢٠١٥ to ١٥/٤/٢٠١٥
- Test and maintenance from ١٥/٤/٢٠١٥ to ١/٦/٢٠١٥
- Documentation from ١/٦/٢٠١٥ to ١٥/٦/٢٠١٥
- Testing ١٥/٦/٢٠١٥ to ٢٥/٦/٢٠١٥

٤.٣ System Analysis

٤.٣.١ System's Characteristics

٤.٣.١.١ Purpose

- Analysis of public opinion to determine the current state of our country.
- How to reveal who plots against our land home.
- To determine who is for or against the current state.
- To know if this Opposition is constructive or destructive to reveal who plots against our land home.
- To contribute in the progress of our land home from this system.

٤.٣.١.٢ Input

Hash tag of twitter that must be classified.

٤.٣.١.٣ Output

The result of the entering hash tag of twitter is the classification of it to be for the current state of our land home or against.

٤.٣.٢ Constrains

- All rules that the administrator follow it.

  Like:-

  - Administrator gives authentication for others.
  - Administrator can generate report.
  - Administrator he can give permission.

٤.٣.٣ Components

- Web site.
- Administrator.

٤.٣.٤ Environment

Web site interface helps administrator in managing system from this interface he can add another one to system, remove some one, gives permission and generate report.

٤.٣.٥ Interrelationship

- Administrator login to his account can add another manager, remove some one from system, gives permission and generate report.

٤.٣.٦ System Boundary

- The system boundary separates it from other systems working in the field.

٤.٣.٧ Interface

- System has interface that enable administrator easily using system, managing it.

٤.٣.٨ System Requirements

The requirements that the system must satisfy are of two types, which are the functional and non-functional requirements. Functional requirements are the requirements that define a function of software that run on the system. Non-Functional requirements are the requirements that specify criteria that can be used to judge the operation of a system, rather than specific behaviors. In the following subsections, both functional and non-functional requirements of the proposed system are listed.

٤.٣.٨.١ Functional Requirements

١. Take data from Twitter Social Network.
٢. Mining this data and then classify it to political or sportive or economic.
٣. If it is political then classify it to be for or to be against.
٤. Draw graph that show that the percentage of agreement or Opposition.

٤.٣.٨.٢ Nonfunctional Requirements

In systems engineering and requirements engineering, a non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors.

This should be contrasted with functional requirements that define specific behavior or functions. The plan for implementing functional requirements is detailed in the system design. The plan for implementing non-functional requirements is detailed in the system architecture.

Broadly, functional requirements define what a system is supposed to do and non-functional requirements define how a system is supposed to be. Functional requirements are usually in the form of "system shall do <requirement>", an individual action of part of the system, perhaps explicitly in the sense of a mathematical function, a black box description input, output, process and control functional model or IPO Model.

In contrast, non-functional requirements are in the form of "system shall be <requirement>", an overall property of the system as a whole or of a particular aspect and not a specific function. The systems' overall properties commonly mark the difference between whether the development project has succeeded or failed.

Non-functional requirements are often called qualities of a system. Other terms for non-functional requirements are "constraints", "quality attributes", "quality goals", "quality of service requirements" and "non-behavioral requirements". Informally these are sometimes called the "ilities", from attributes like stability and portability. Qualities, that are non-functional requirements, can be divided into two main categories:

- Execution qualities, such as security and usability, which are observable at run time.
- Evolution qualities, such as testability, maintainability, extensibility and scalability, which are embodied in the static structure of the software system.

Accessibility: that the system should be easy to access. This achieved by,

١. Application can be easily used from the user-friendly GUI.

Accuracy: that the system should provide accurate results. This achieved by,

- Mining data and classify it to ensure that the result is accurate.

Security: that the system should be able to protect the data and resources. This achieved by,

- Only few of people use this system so security is very high.
- Everyone has his authentication to use this system.
- There is a general manager that gives permission to others.
- Only managers see the generated report.

Tools requirements

- Eclipse
- E-draw max
- SPARQLquery
- Jena---A Java API for RDF
- JDK
- Protégé
- Tomcat

If you want to study RDF, first you should have a basic understanding of the following.

- HTML
- XHTML
- XML
- XML Namespace

٤.٤ System Design

٤.٤.١ Analysis Use Case

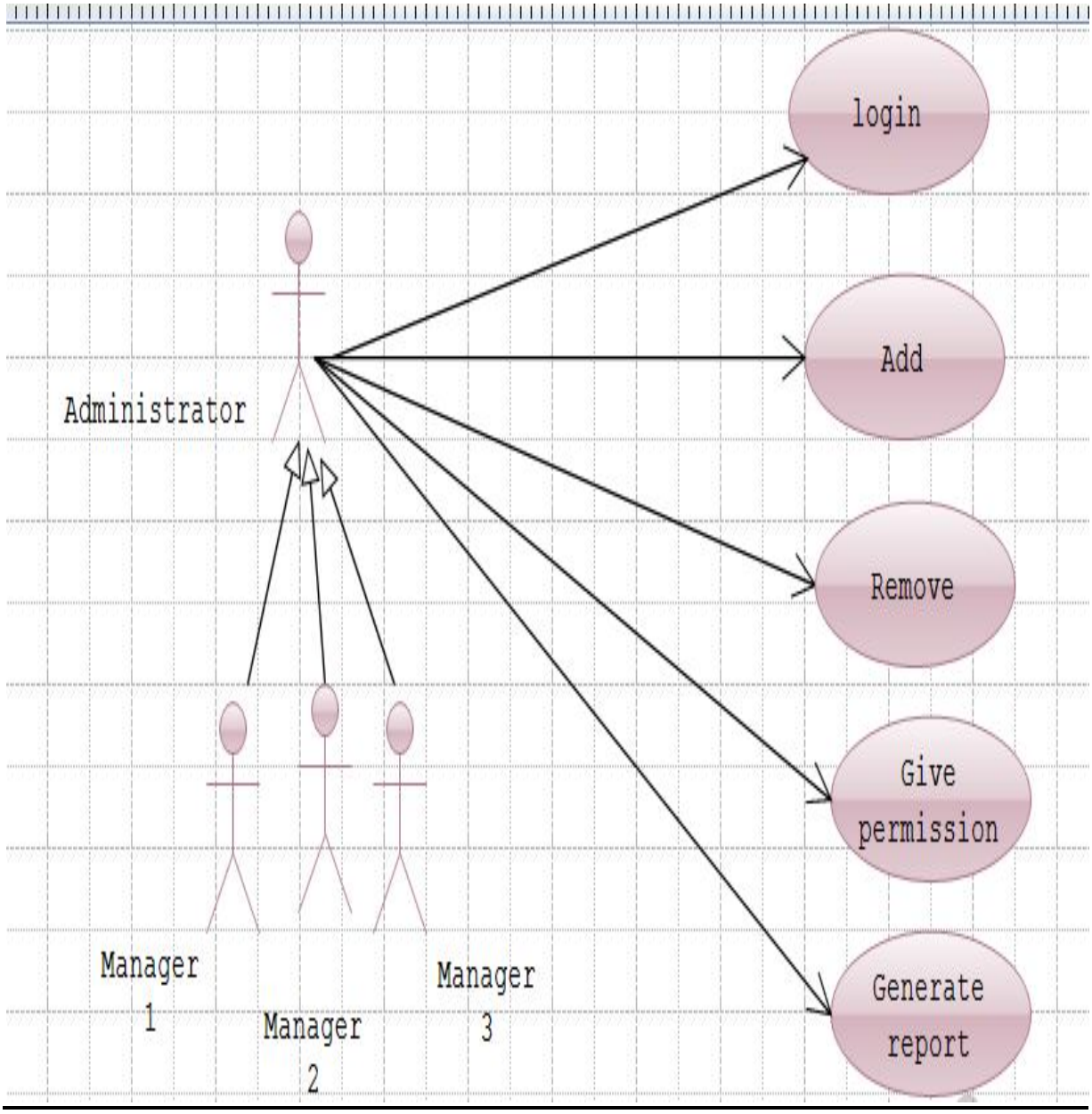


Figure ٤.٤.١ use case diagram

Use Case Name:	Login	
Actors	Administrator	
Description	This use case describes the process that an administrator can login using his user name and password to this system	
Typical Course of Events	Actor Action	**System Response**
	Step١:this use case is initiated only when Administrator send request to enter web site. Step٢ the administrator submit his user name and password.	 Step٣ the administrator information such as user name and password are valid Step٤ confirm and successfully login to the web site.
Alternate Courses:	Administrator's information is not valid return error message	
Precondition:	Administrator must has account	
Post condition:	None.	
Assumption:	None at this time.	

٤.٤ table login use case

Use Case Name:	Add manager	
Actors	Administrator	
Description	This use case describes the process that an administrator can add another administrator under his leadership	
Typical Course of Events	Actor Action	**System Response**
	Step١: this use case is initiated only when Administrator send request to add another person to the system. Step٣ the administrator add information of new administrator. Step ٤ administrator clicks ok to submit information.	**Step٢** adding form will display to enter information of administrator will be added. **Step٥:** confirm that information is successfully saved in database.
Alternate Courses:	None	
Precondition:	Administrator must login to the system	
Post condition:	New administrator can use system.	
Assumption:	None at this time.	

٤.٤ table for adding new person

Use Case Name:	Remove manager	
Actors	Administrator	
Description	This use case describes the process that an administrator can remove another administrator under from system.	
Typical Course of Events	Actor Action	**System Response**
	Step١: this use case is initiated only when Administrator send request to remove another person from the system. Step٣ the administrator add information of administrator needed to be removed. Step ٤ administrator clicks ok to submit information.	**Step٢** removing form will display to enter information of administrator that will be removed. **Step٥:** confirm that information is successfully removed from database.
Alternate Courses:	None	
Precondition:	Administrator must login to the system	
Post condition:	This removed administrator cannot use system.	
Assumption:	None at this time.	

٤.٤ table for removing some person from system

Use Case Name:	Give permission	
Actors	Administrator	
Description	This use case describes the process that an administrator can permission to another administrators who are under his leadership.	
Typical Course of Events	Actor Action	**System Response**
	Step١: this use case is initiated only when Administrator send request to add some permission to others. Step٣ the administrator add rules. **Step٤:** administrator choose person that rules will be assigned for them. **Step٥** administrator clicks ok to submit rules.	**Step٢** rules form will display to enter rules that will be assigned. **Step٦:** confirm that rules are successfully submitted to certain administrator.
Alternate Courses:	None	
Precondition:	Administrator must login to the system	
Post condition:	Administrator who rules assigned to him has certain function to perform according to these rules.	
Assumption:	None at this time.	

٤.٤ table for giving permission

Use Case Name:	Generate report	
Actors	Administrator	
Description	This use case describes the process that an administrator can generate report.	
Typical Course of Events	Actor Action	**System Response**
	Step١: this use case is initiated only when Administrator send request for report.	**Step٢** web server will generate report. **Step٣:** web server will display report in window.
Alternate Courses:	None	
Precondition:	Administrator must login to the system	
Post condition:	None	
Assumption:	None at this time.	

٤.٤ table for generating report

٤.٤.٢ Sequence Diagram:

١. Log in:

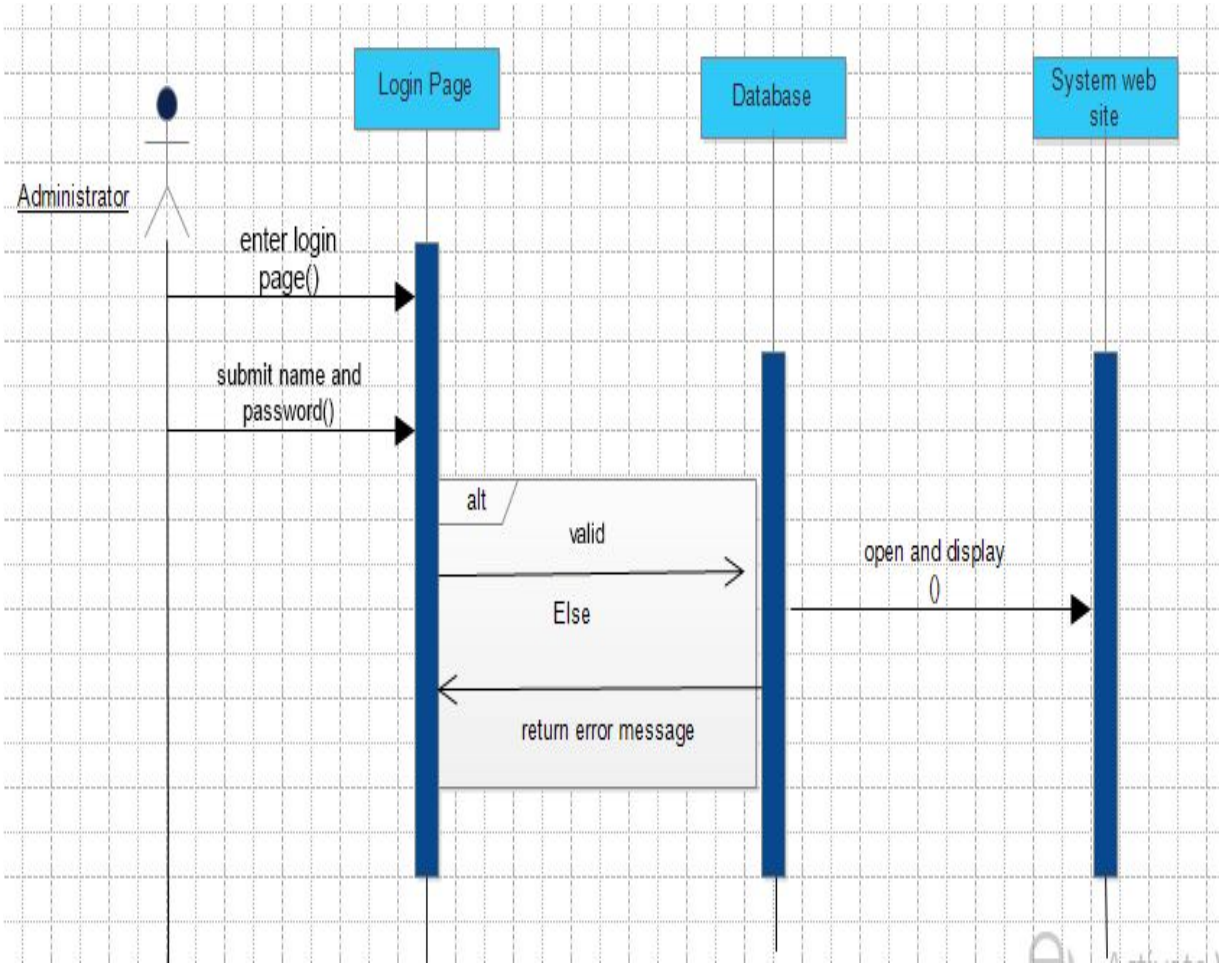


Figure ٤.٤.٢ sequence of login

٢. Add new manager

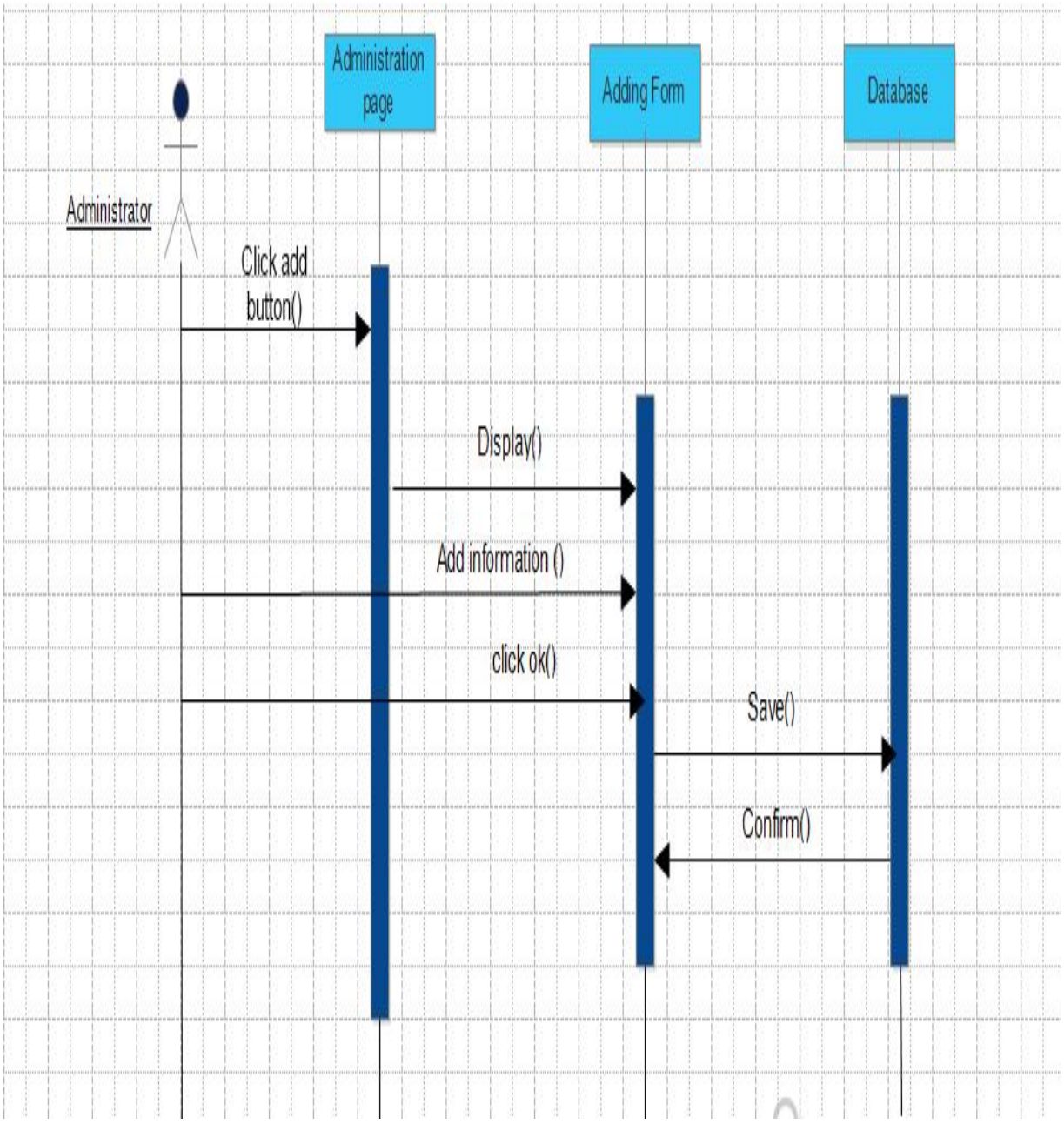


Figure ٤.٤.٢ sequence of adding manager

٣. Remove another administrator

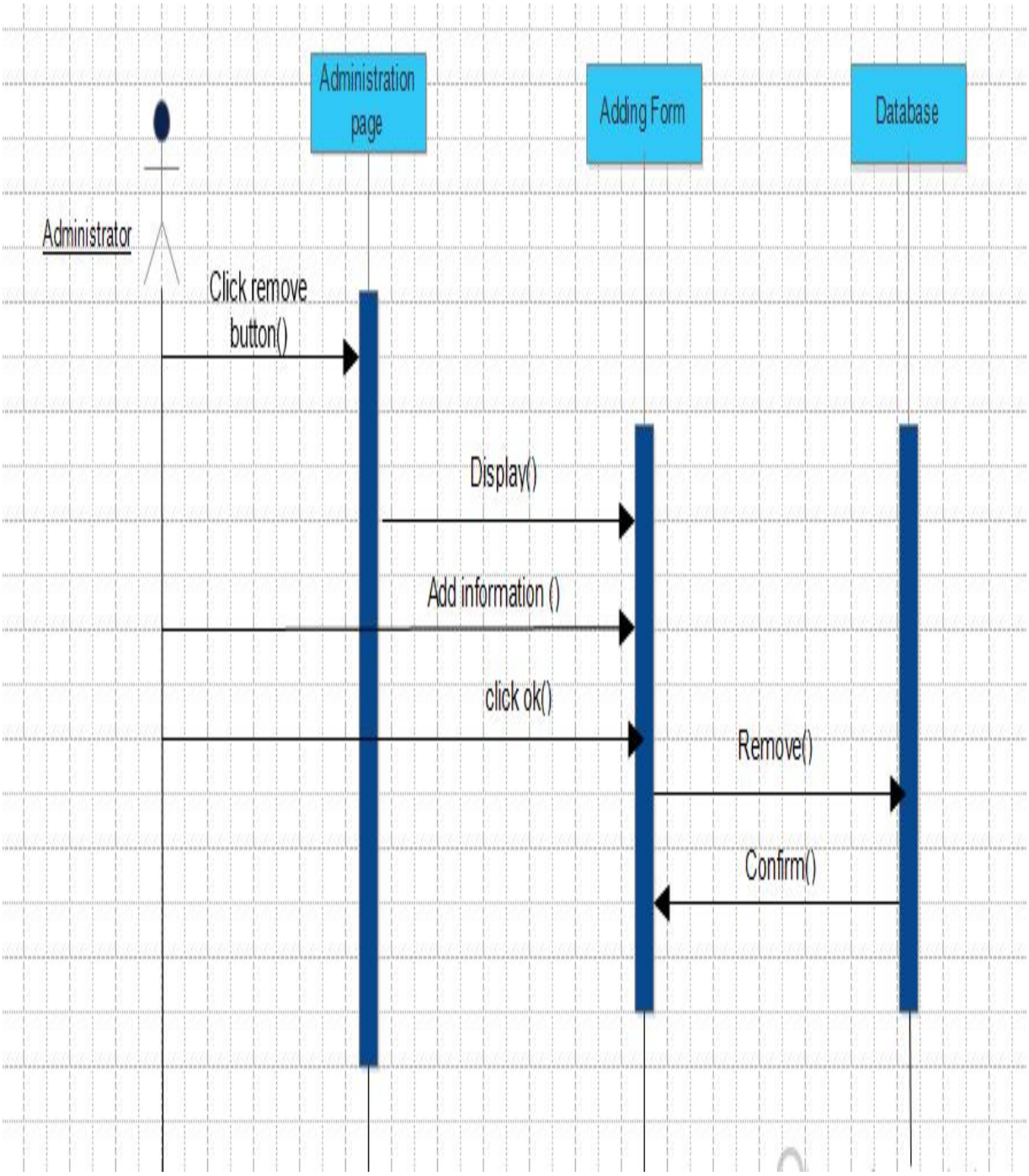


Figure ٤.٤.٢ Remove

٤. Give permission

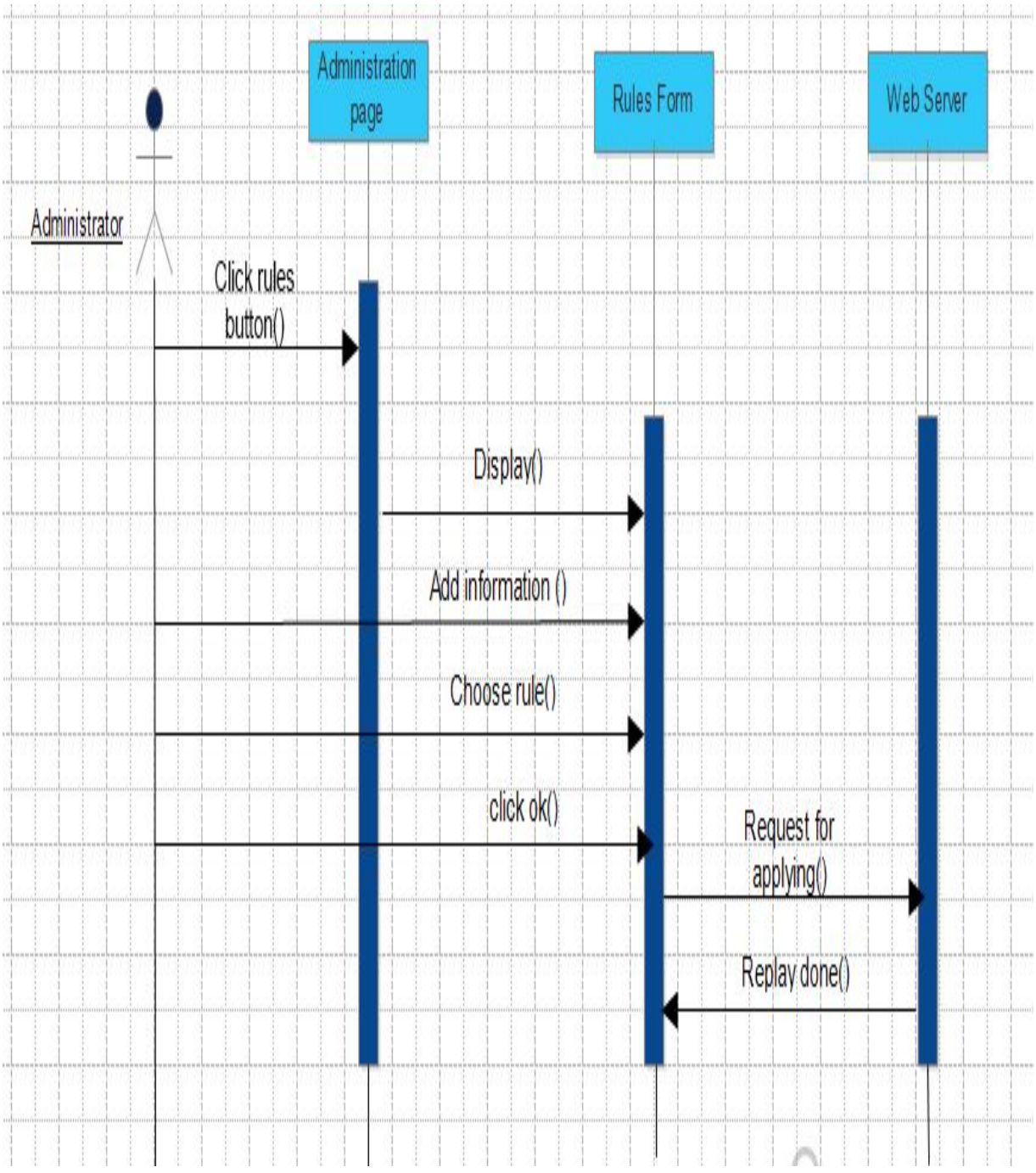


Figure ٤.٤.٢ sequence of giving permission

٥. Generate report

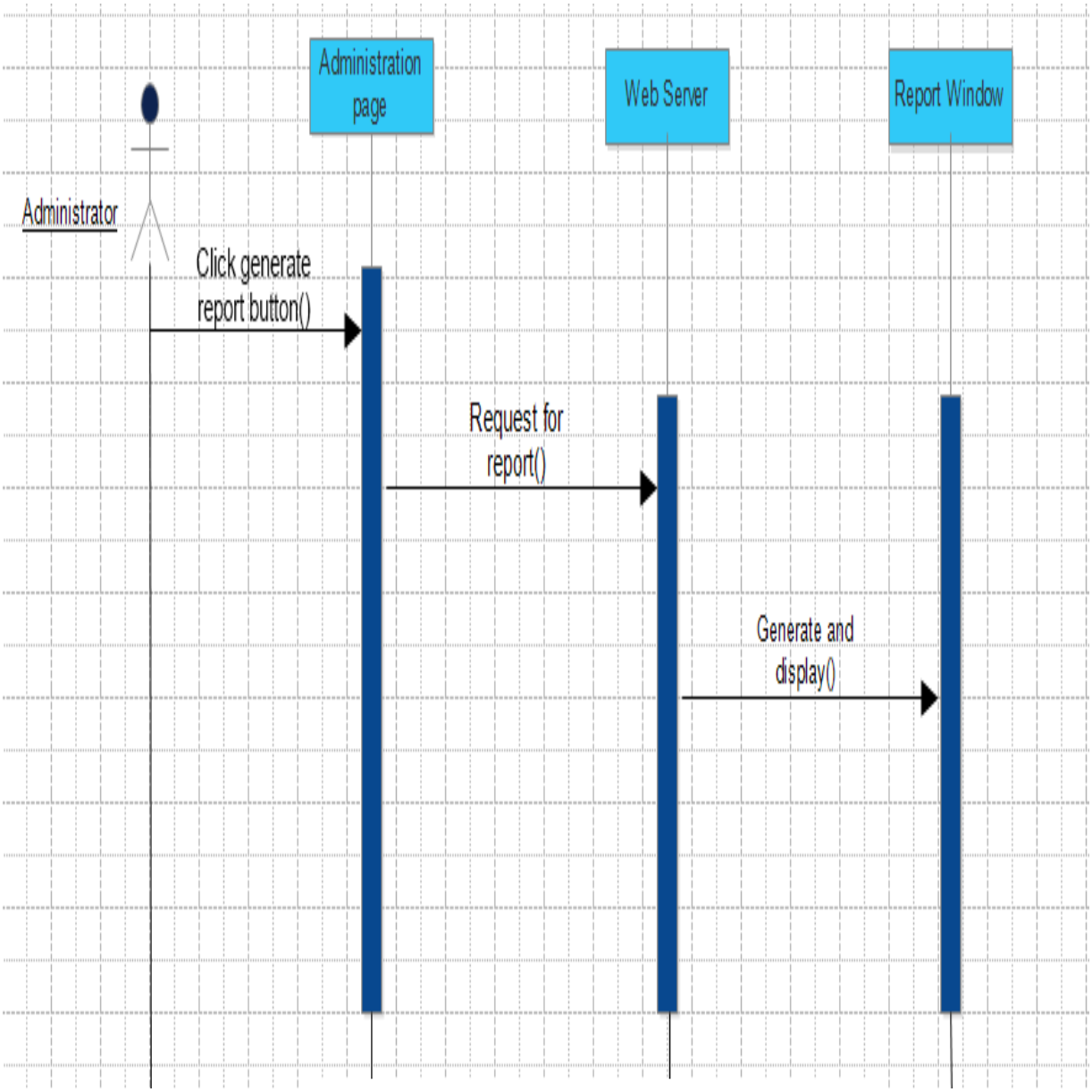


Figure ٤.٤.٢ sequence for generating report

٤.٤.٣ Collabortion Diagram

١. Add another administrator

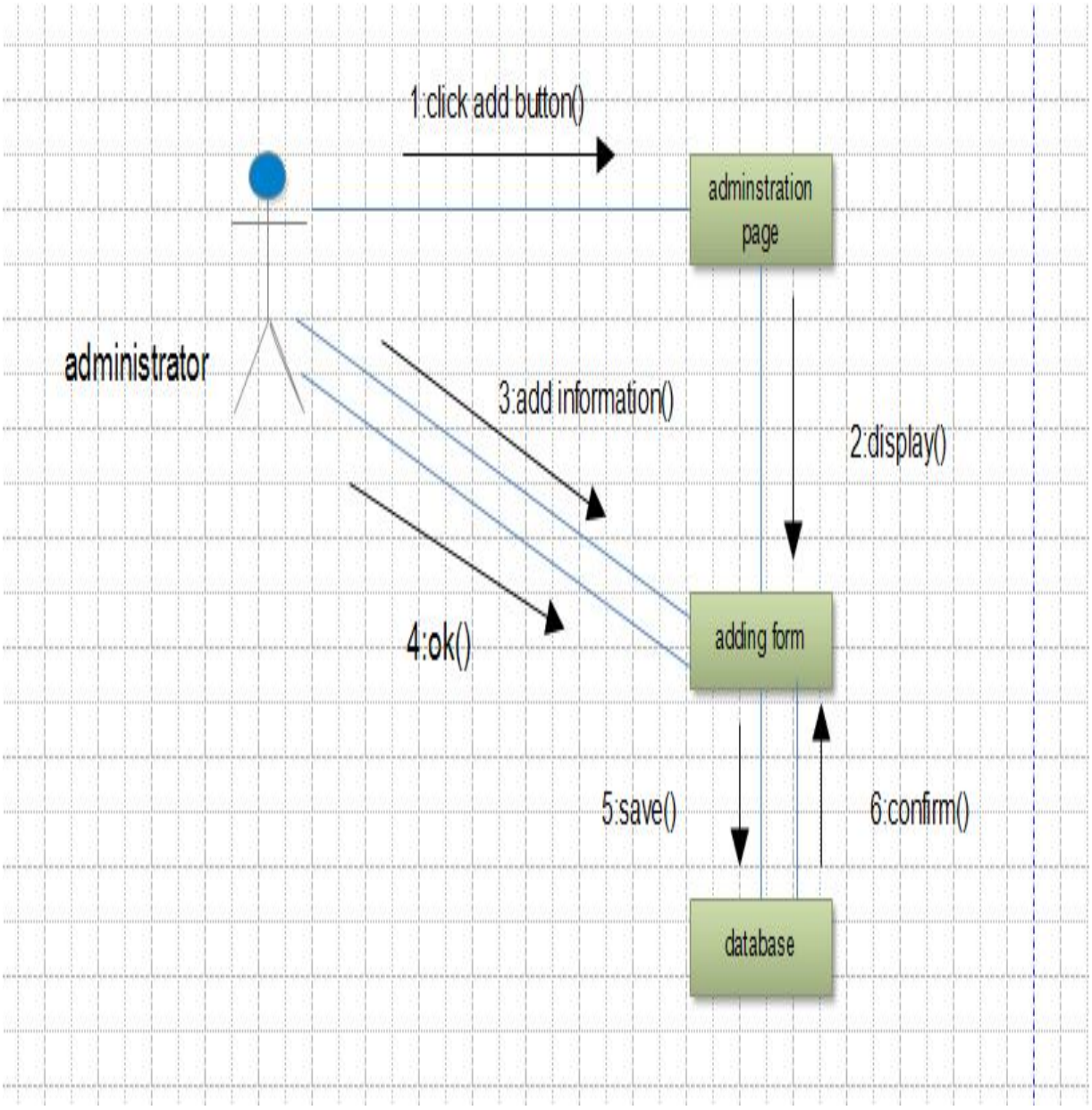


Figure ٤.٤.٣ Collabortion for adding

٢. Removing another administrator

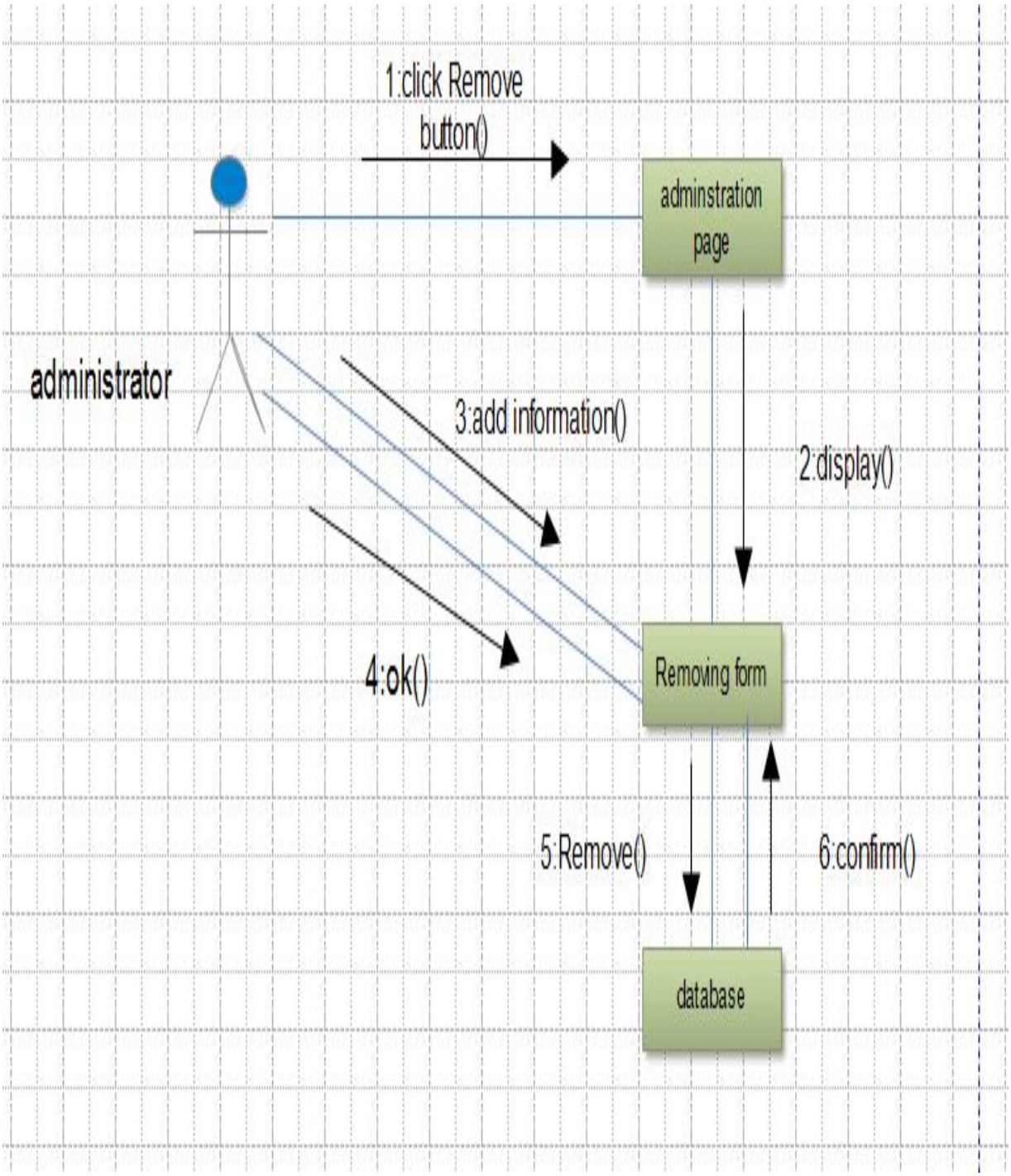


Figure ٤.٤.٣ Collabortion for removing

٣. Give permission

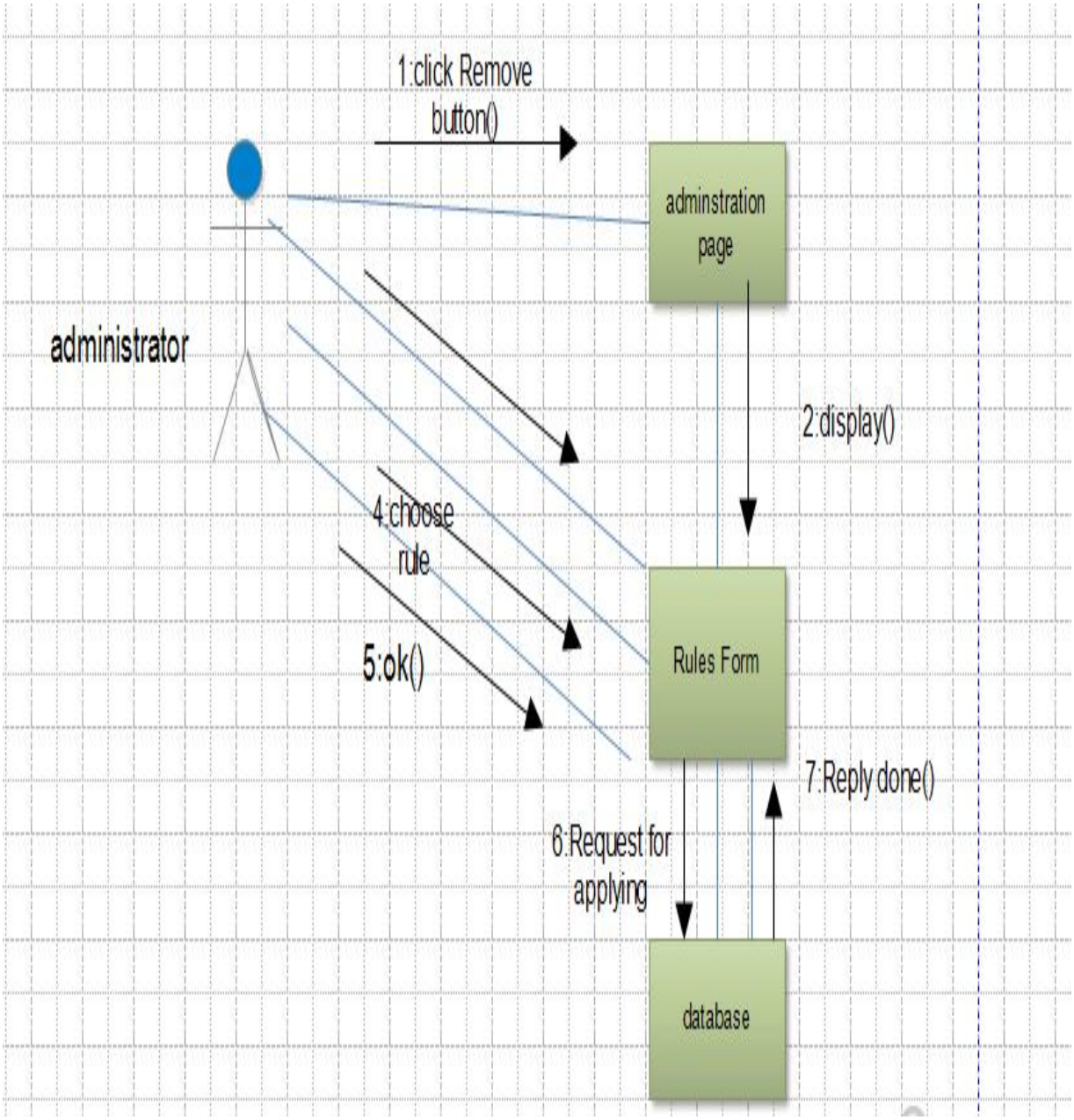


Figure ٤.٤.٣ Collabortion for giving permission

٤. Generate report

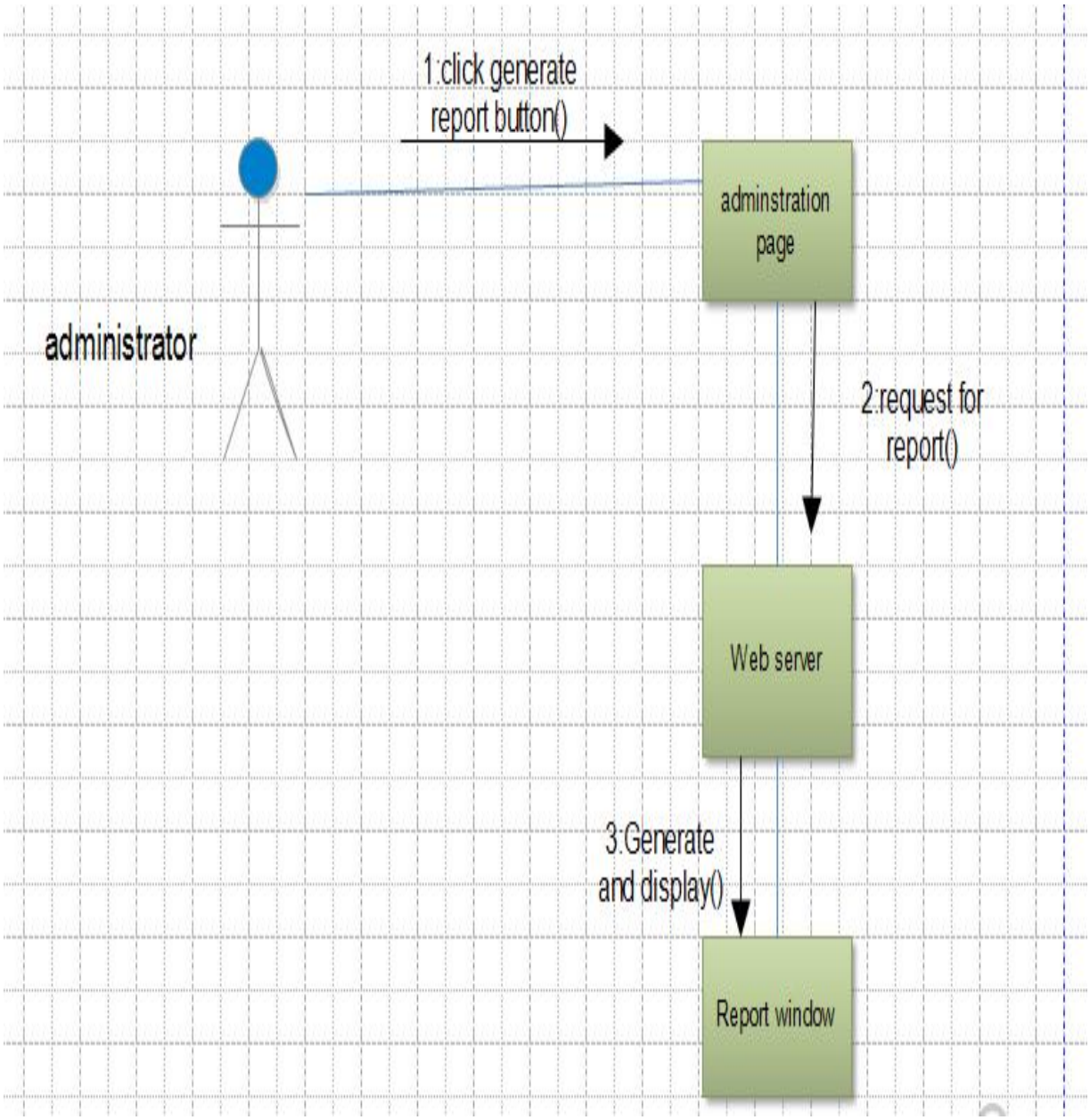


Figure ٤.٤.٣ Collabortion for generating report

٤.٤.٤ Activity Diagram

١. Login

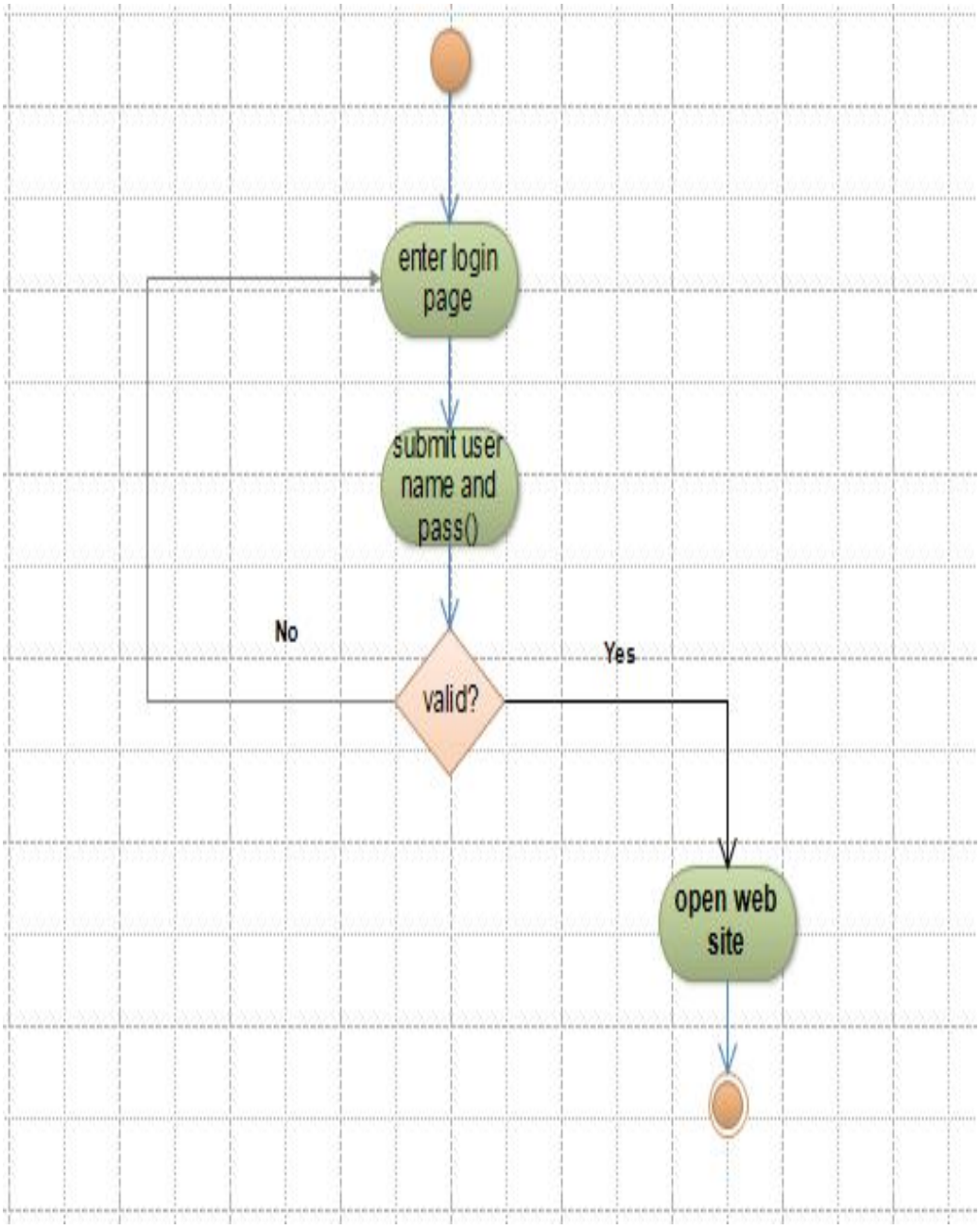


Figure ٤.٤.٤ activity for login

٢. Add

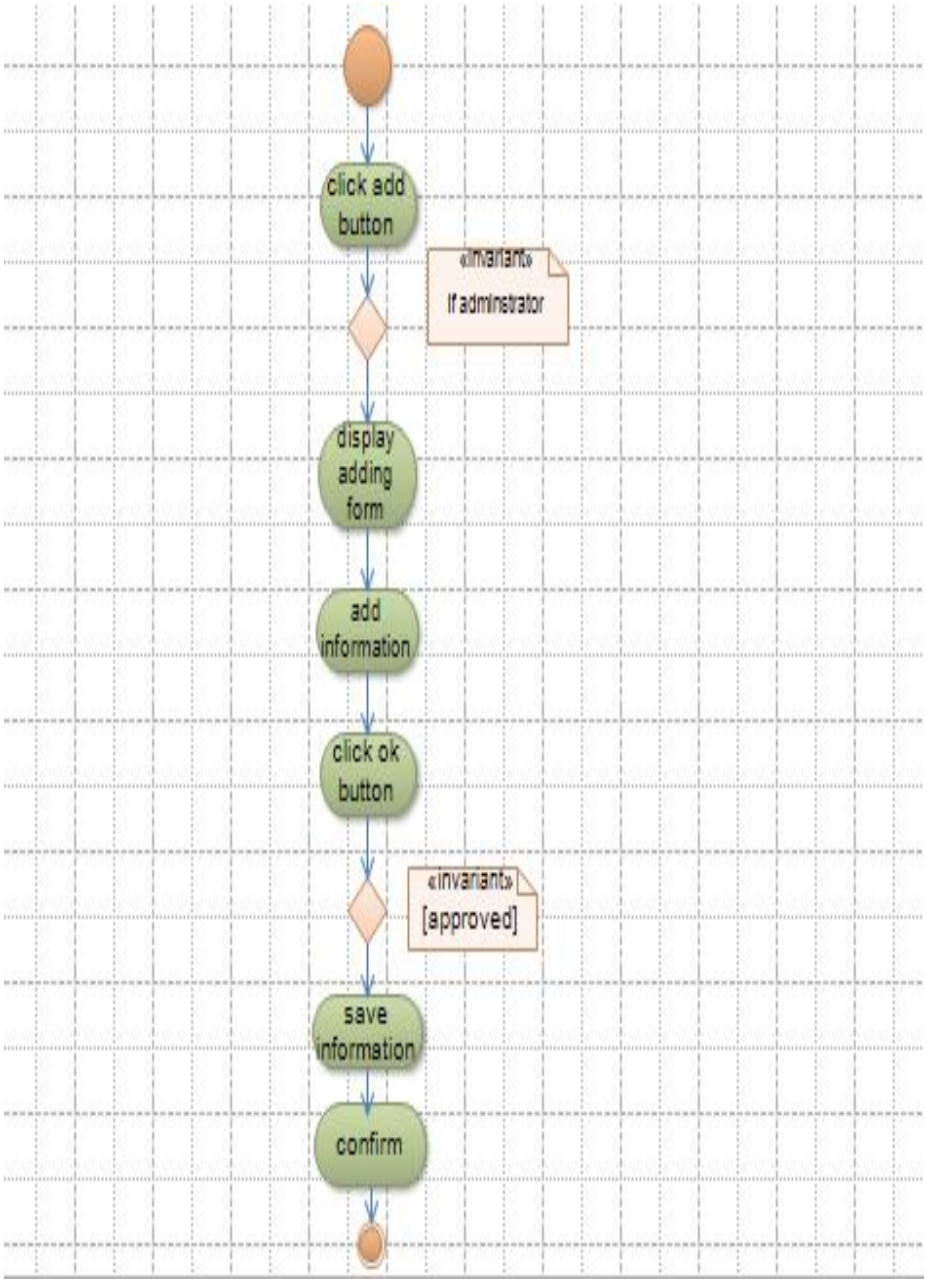


Figure ٤.٤.٤ activity for adding

٣. Remove

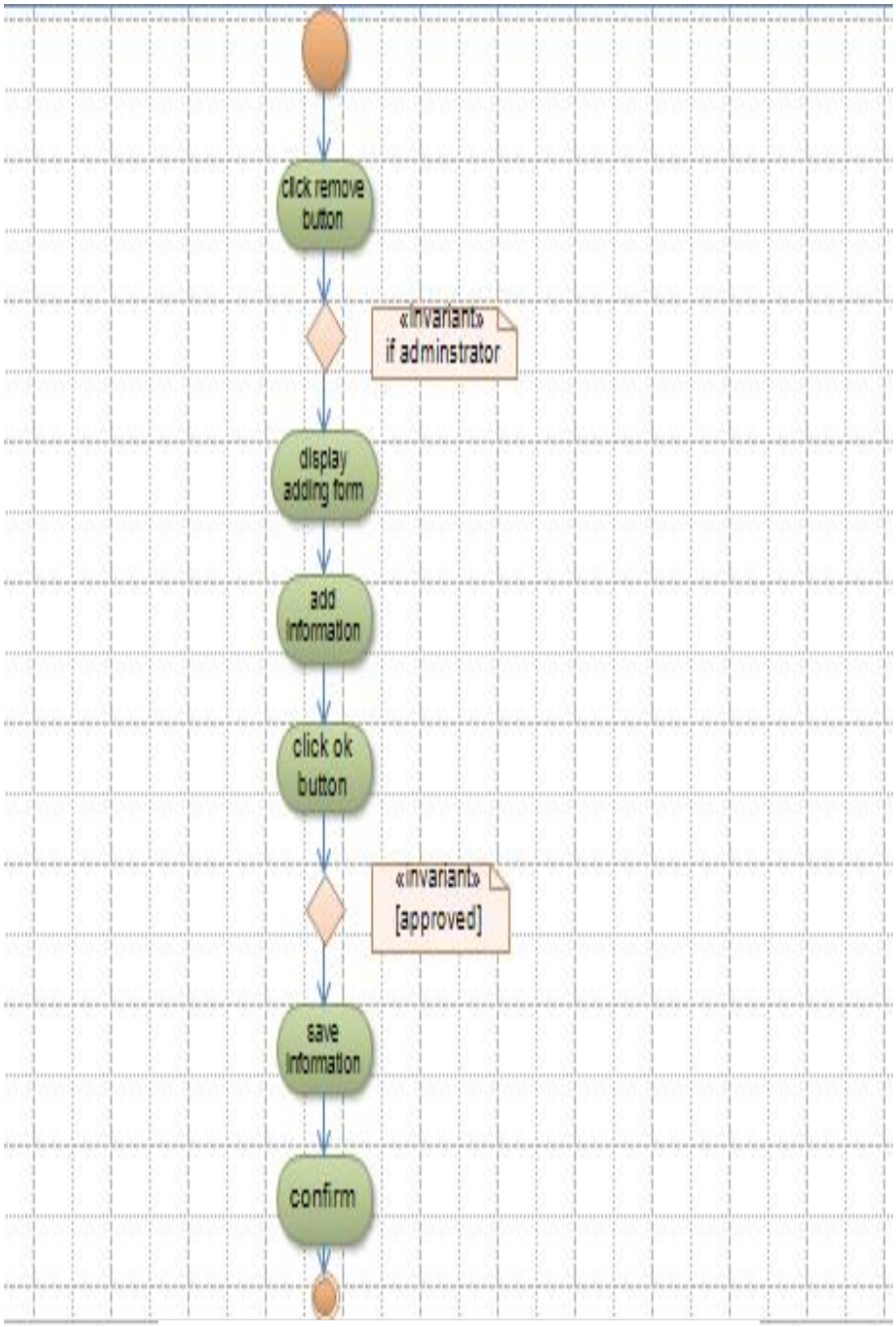


Figure ٤.٤.٤ activity for removing

٤. Give permission

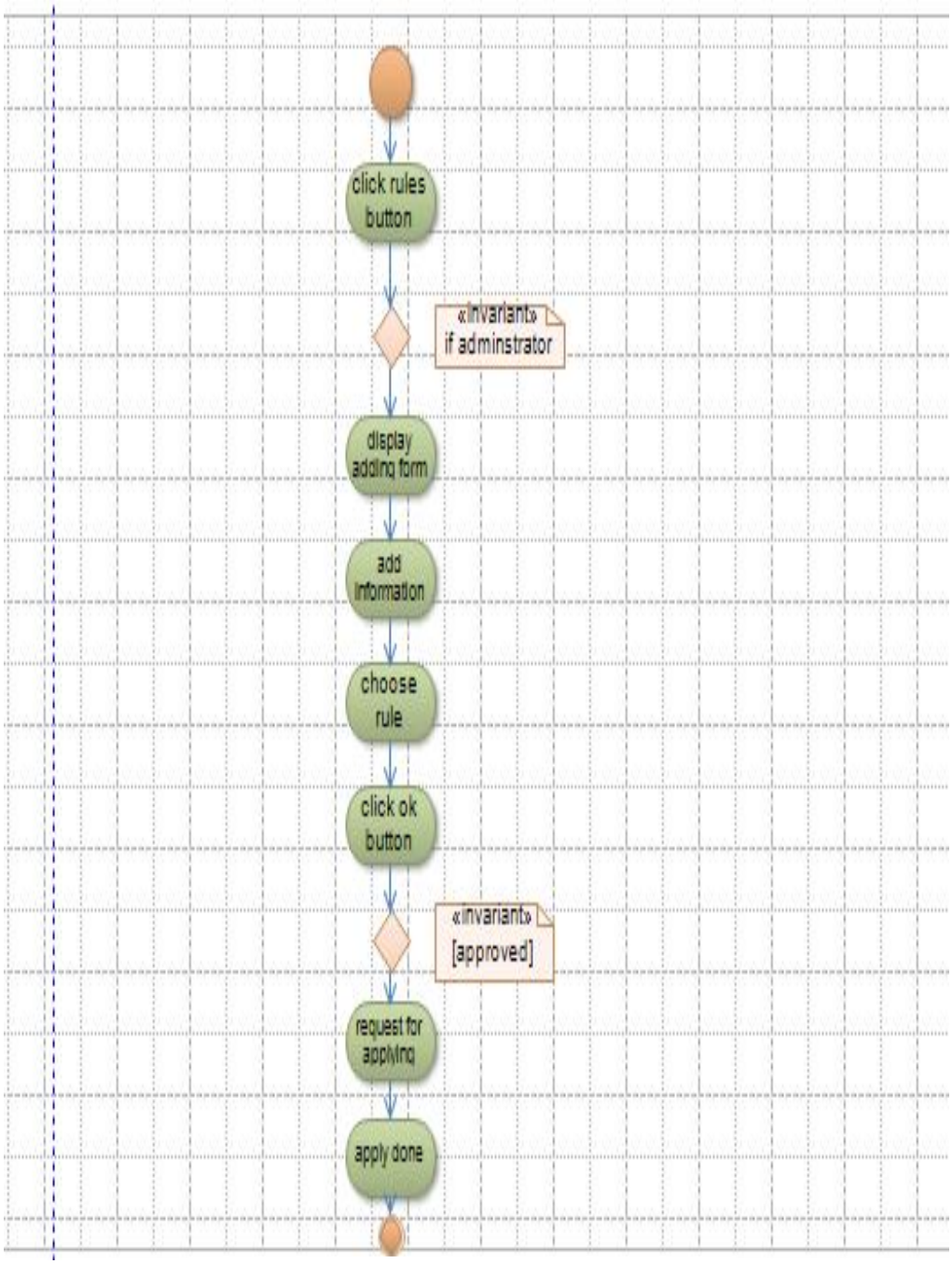


Figure ٤.٤ activity for giving permission

٥. Generate report

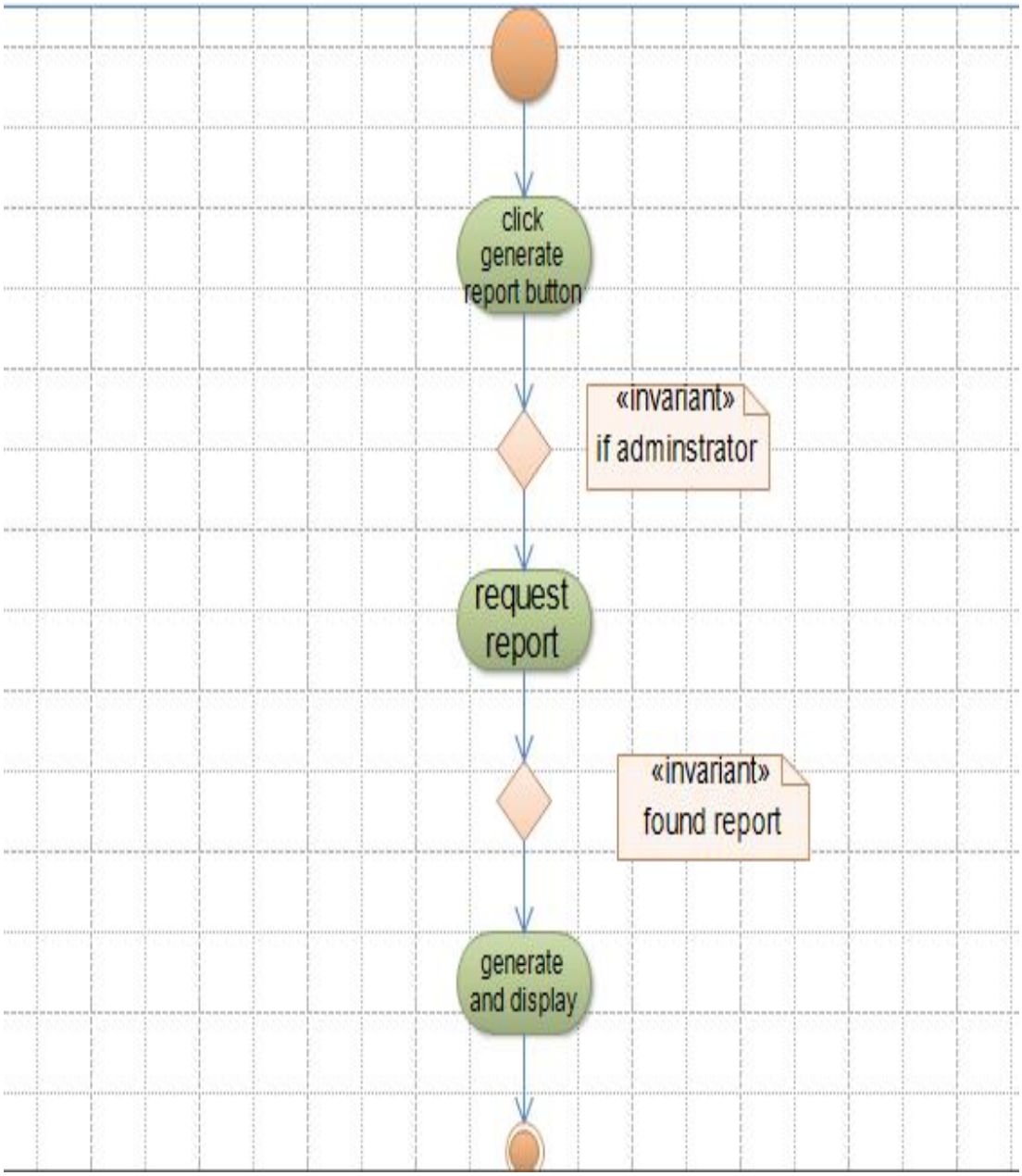


Figure ٤.٤.٤ activity for generating report

٤.٤.٥ Class Diagram

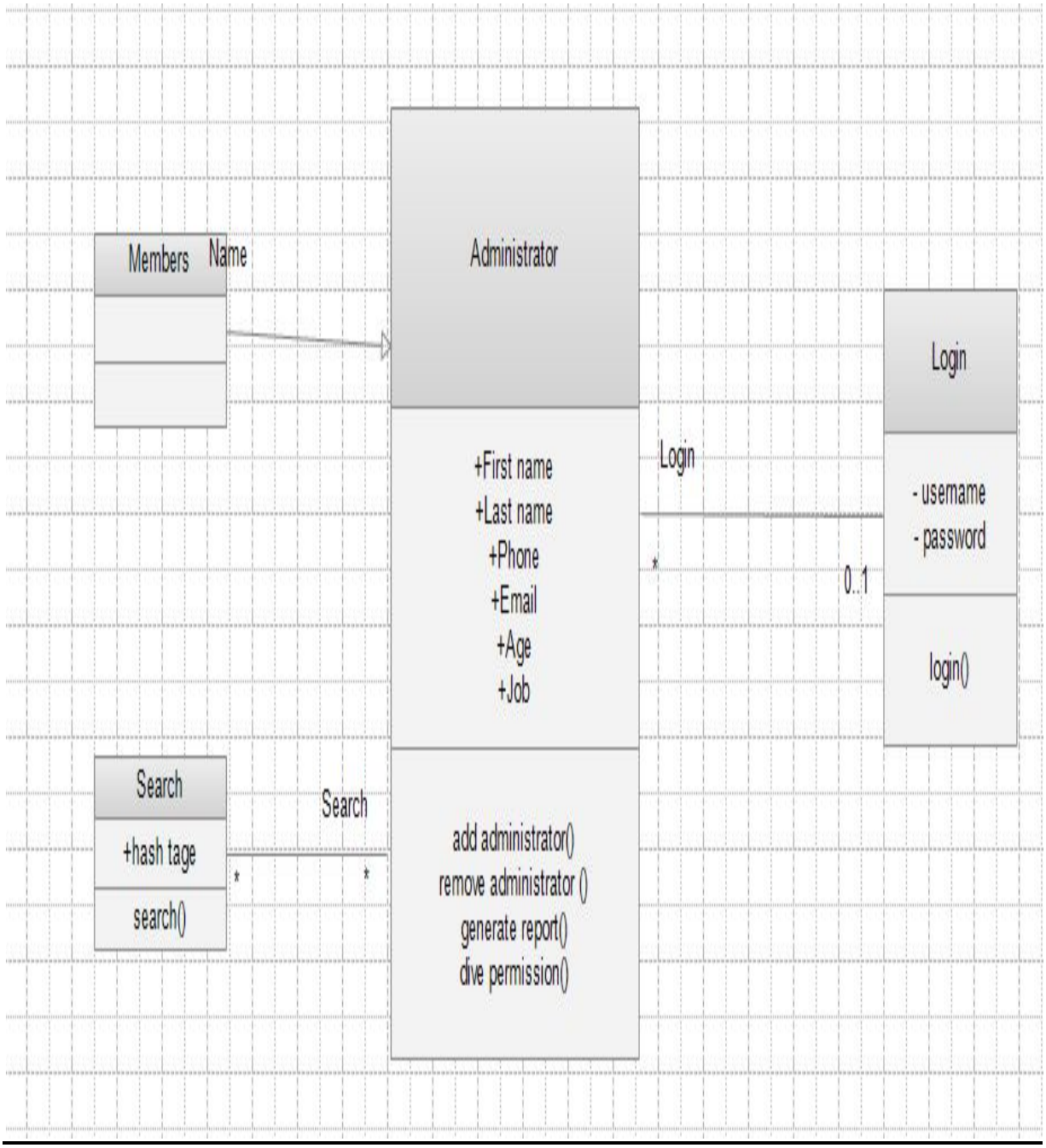


Figure ٤.٤.٥ Class Diagram

٤.٥ System Implementation

The implementation phase is the pre-final phase of the SDLC and it involves the actual construction and installation of the system. This phase also includes the maintenance of the system and any future updates or expansion of the system.

With the database project example, the following activities would be common:

- Physical installation of protégé editor V.٤.٣ for creating ontology.
- Installation of Eclipse and JDK using Java EE.
- Installation of Tomcat for creating JSP.
- Adding libraries of Jena framework for building Semantic Web application.
- Using SPARQL Query for retrieving data from Ontology.
- Initialization and continuous operation of the ontology.
- Maintaining the hardware and software hosting the ontology.
- Any future expansion of the ontology.

This phase is the longest phase as it has no defined endpoint, with the exception of the end of the system and its users (Amazon shuts down, Google calls it quits, etc.).

The implementation phase should make very few changes to the system design, although some changes are inevitable. It is critical for maintainability that all changes be incorporated back into the design specifications. Otherwise, the value of the system design will be lost as soon as the system is released, and the only design will be the code.

A system documented only by its code is very hard for management to understand and upgrade, outsource, or disseminate. The implementation phase is operating with a server handicap, if not doomed to failure. Domain expertise must remain within the project throughout its life cycle. The dilemma is that once up-front analysis is complete, the analysis has less work, and their role becomes more passive.

Often this cannot be justified, and these people are valuable to the business and are needed elsewhere. A solution that often works well is to keep a few domain experts assigned full time, and give them the permanent role of facilitator. In this capacity, they perform domain analysis, and execute all change requests to requirements specifications.

They also develop user-oriented test plans, and construct system documentation. Their role therefore remains an active one, and their knowledge about the application, and contacts within the organization can still be tapped when questions arise during development.

٤.٦ Maintenance

Maintenance includes all the activity after the installation of software that is performed to keep the system operational.

As we have mentioned earlier, software often has designed faults. The two major forms of maintenance activities are adaptive maintenance and corrective maintenance.

It is generally agreed that for large systems, removing all faults before delivery is extremely difficult and faults will be discovered long after the system is installed. As these faults are detected, they have to be removed.

Maintenance activities related to fixing of errors fall under corrective maintenance.

Removing errors is one of the activities of maintenance. Maintenance also needed due to a change in the environment or the requirements of the system. The introduction of a software system affects the work environment.

This change in environment often changes in the desired from the system. Furthermore, often after the system is installed and the users have had a chance to work with it for some time, requirements that are not identified during requirement analysis phase will be uncovered. This occurs, since the experience with the software helps the user to define the needs more precisely.

There might also be changes in the input data, the system environment and output formats. All these require modification of the software. The maintenance activities related to such modification fall under adaptive maintenance.

Maintenance work is based on existing software, as compared to development work, which creates new software.

Consequently maintenance resolves around understanding the existing software and spares most of time trying to understand the software that they have to modify. Understanding the software involves not only understanding the code, but also the related documents.

During the modification of the software, the effects of the change have to be clearly understood by the maintainer since introducing undesired side effects in the system during modification is easier.

To test whether those aspects in the system that are not supposed to be modified are operating as they were before modification, regression testing is done.

Regression testing involves executing old test cases to test that no new errors have been introduced.

Thus, maintenance involves understanding the existing software (code and related documents), understanding the effects of changes, making the change both to the code and document, testing the new parts (changes), and resetting of the old parts that were not changed.

Since often during development, needs of the maintainers are not kept in mind, little support documents are produced during development to aid the maintainer.

The complexity of the maintenance task is coupled with the neglect of maintenance concerns during development which makes maintenance the most cost effective activity in the life of software product.

Chapter 5
Data Mining

٥.١ Overview

٥.٢ Data, Information and Knowledge

٥.٣ What can data mining do?

٥.٤ How does mining work?

٥.٥ Stemming

٥.٦ Data mining in “System Analysis of Public Opinion”

٥.٧ Twitter Data Collector

٥.٨ Conclusion

Generally, data mining (sometimes called data or knowledge discovery) is the process of analyzing data from different perspectives and summarizing it into useful information - information that can be used to increase revenue, cuts costs, or both.

Data mining software is one of a number of analytical tools for analyzing data. It allows users to analyze data from many different dimensions or angles, categorize it, and summarize the relationships identified.

Technically, data mining is the process of finding correlations or patterns among dozens of fields in large relational databases.

Continuous Innovation

Although data mining is a relatively new term, the technology is not. Companies have used powerful computers to sift through volumes of supermarket scanner data and analyze market research reports for years.

However, continuous innovations in computer processing power, disk storage, and statistical software are dramatically increasing the accuracy of analysis while driving down the cost

٥.٢ Data, Information and Knowledge

٥.٢.١ Data

Are any facts, numbers, or text that can be processed by a computer. Today, organizations are accumulating vast and growing amounts of data in different formats and different databases. This includes:

- Operational or transactional data such as, sales, cost, inventory, payroll, and accounting.
- nonoperational data, such as industry sales, forecast data, and macro-economic data
- meta data - data about the data itself, such as logical database design or data dictionary definitions

٥.٢.٢ Information

The patterns, associations, or relationships among all this data can provide information. For example, analysis of retail point of sale transaction data can yield information on which products are selling and when.

٥.٢.٣ Knowledge

Information can be converted into knowledge about historical patterns and future trends. For example, summary information on retail supermarket sales can be analyzed in light of promotional efforts to provide knowledge of consumer buying behavior.

Thus, a manufacturer or retailer could determine which items are most susceptible to promotional efforts.

٥.٣ What can data mining do?

Data mining is primarily used today by companies with a strong consumer focus - retail, financial, communication, and marketing organizations. It enables these companies to determine relationships among "internal" factors such as price, product positioning, or staff skills, and "external" factors such as economic indicators, competition, and customer demographics.

And, it enables them to determine the impact on sales, customer satisfaction, and corporate profits. Finally, it enables them to "drill down" into summary information to view detail transactional data.

With data mining, a retailer could use point-of-sale records of customer purchases to send targeted promotions based on an individual's purchase history. By mining demographic data from comment or warranty cards, the retailer could develop products and promotions to appeal to specific customer segments.

٥.٤ How does data mining work?

While large-scale information technology has been evolving separate transaction and analytical systems, data mining provides the link between the two. Data mining software analyzes relationships and patterns in stored transaction data based on open-ended user queries.

Several types of analytical software are available: statistical, machine learning, and neural networks. Generally, any of four types of relationships are sought:

- **Classes:** Stored data is used to locate data in predetermined groups. For example, a restaurant chain could mine customer purchase data to determine when customers visit and what they typically order. This information could be used to increase traffic by having daily specials.
- **Clusters:** Data items are grouped according to logical relationships or consumer preferences. For example, data can be mined to identify market segments or consumer affinities.
- **Associations:** Data can be mined to identify associations. The beer-diaper example is an example of associative mining.
- **Sequential patterns:** Data is mined to anticipate behavior patterns and trends. For example, an outdoor equipment retailer could predict the likelihood of a backpack being purchased based on a consumer's purchase of sleeping bags and hiking shoes.

In our project Data Mining has a great role in filtering tweets from Twitter using classification technique that Stored data is used to locate data in predetermined groups to know political opinions and non-political such as sports , social, religious, but before classification we have stem process what is stemmer ?

٥.٥ Stemming

Stemming is the term used in linguistic morphology and information retrieval to describe the process for reducing inflected (or sometimes derived) words to their word stem, base or root form generally a written word form.

The stem needs not to be identical to the morphological root of the word; it is usually sufficient that related words map to the same stem, even if this stem is not in itself a valid root.

Algorithms for stemming have been studied in computer science since the ١٩٦٠s. Many search engines treat words with the same stem as synonyms as a kind of query expansion, a process called conflation.

٥.٦ Data Mining in “System Analysis of Public Opinion”

Data mining play an important role in this system, it seems to be the first step in this system it goes in many steps.

١. The first step is to extract key words and remove stop words and special characters.

```
public class key_words_extractor extends JFrame implements ActionListener{

public key_words_extractor(String KeyWordsFile,String KeyWordsCount){

    wordsfile=KeyWordsFile;

    countfile=KeyWordsCount;

    try{

       ShowBrogram();


    }catch(Exception e){

       System.err.println("Error"+ e.getMessage());

    }

  }

  ////////////////////////////////////////////////////////////////

public void ShowBrogram(){

    setSize(٤٨٠, ٤٠٠);

    setLocation(١٠,١٠);
```

```
        setVisible(true);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setTitle("Get your key words");
        pform.setLayout(null);
        pform.setSize(this.getWidth(),this.getHeight());
        pform.setBackground(Color.black);
//              Buttons             //
        bBrows .setBounds(٣٨٠, ١٥, ١٠٠, ٢٠);
        bBrows.addActionListener(this);
        pform.add(bBrows);
        bkeywords .setBounds(٣٨٠, ٤٥, ١٠٠, ٢٠);
        bkeywords.addActionListener(this);
        pform.add(bkeywords);
        bopen .setBounds(٣٨٠,٧٥, ١٠٠, ٢٠);
 bopen.addActionListener(this);
         //              scrollbar.....//
        SbeforStem.setBounds(١٠, ٦٥, ١٦٥, ٢٥٠);
        SbeforStem.getViewport().add(Tbeforstem);
        pform.add(SbeforStem);
        SafterStem.setBounds(٢٠٠, ٦٥, ١٦٥, ٢٥٠);
        SafterStem.getViewport().add(Tafterstem);
```

```
        pform.add(SafterStem);
//                text            //
        tpass.setBounds(١٠, ١٥, ٣٤٠, ٢٠);
        tpass.setForeground(Color.blue);
        pform.add(tpass);
        Tbeforstem.setForeground(Color.blue);
        Tafterstem.setForeground(Color.black);
        //              label             //
        lbeforStem.setBounds(١٠, ٤٥,١٥٠, ٢٠);
        lbeforStem.setForeground(Color.white);
        pform.add(lbeforStem);
        lafterStem.setBounds(٢٠٠, ٤٥,١٥٠, ٢٠);
        lafterStem.setForeground(Color.white);
        pform.add(lafterStem);
        pform.add(bopen);
        add(pform);
    }

    @Override
public void actionPerformed(ActionEvent e) {
//            choose file           //
```

```
if(e.getSource()==bBrows)
try{
            JFileChooser DialogFile=new JFileChooser();
             DialogFile.setFileSelectionMode(JFileChooser.FILES_ONLY);
            int selectedFile=DialogFile.showOpenDialog(null);
if(selectedFile==JFileChooser.CANCEL_OPTION)
return;
            tpass.setText(DialogFile.getSelectedFile().toString());
          }catch(Exception ee){
          System.err.println("Error"+ ee.getMessage());
          }
//             open file          //
       if(e.getSource()==bopen)
          try{
            File openFile =new File(tpass.getText());
            Desktop.getDesktop().open(openFile);
          }catch(Exception ee){
          System.err.println("Error"+ ee.getMessage());
          }
       //   get key words on file //
if(e.getSource()==bkeywords)
```

```
try{
            String [] myfileInwords=new String[٢٠٠٠٠];
            String myFile="",fileAsString="";
            myFile=tpass.getText();
            //change file into string to start working on it
            fileAsString=ChangeFiletoString(tpass.getText());
            //change your string to lower case
            fileAsString=fileAsString.toLowerCase();
// to delete sptial characters and numbers from your string //
char[]                                          ch={'١','٢','٣','٤','٥','٦','٧','٨','٩','٠','-
','_','!','%',',','.','@','*','&',')','(',';',"","=','•','/','#'};
for(int i=٠;i<ch.length;i++){
              fileAsString=fileAsString.replace(ch[i], ' ');}
            // change your string to array of words //
              myfileInwords=fileAsString.split("");
             // delete stop words from my file    //
// note: reuse the string fileAsString to work
             //on stop words file to save memory
              fileAsString=ChangeFiletoString("stopwords.txt");
              fileAsString=fileAsString.toLowerCase();
              String [] stopWordsInwords=fileAsString.split(" ");
              String [] myfileWithoutStop=new String[٢٠٠٠٠];
```

```
            String[] wordsForStem=new String[٢٠٠٠٠];
            int numOfwords=٠;
 for(int t=٠;t<myfileInwords.length;t++)
for(int b=٠;b<stopWordsInwords.length;b++)
                  //replace each stop word in your file by *
if(myfileInwords[t].equals(stopWordsInwords[b]))
                     myfileInwords[t]="*";
for(int u=٠;u<myfileInwords.length;u++)
            {if(myfileInwords[u].equals("*"))
continue;
else
               {myfileWithoutStop[numOfwords]=myfileInwords[u];
               wordsForStem[numOfwords]=myfileInwords[u];
               numOfwords++;}
               }

//          stemming    //
            String Arabicfile="";
for(int t=٠;t<numOfwords;t++)
            Arabicfile+=wordsForStem[t]+"\n";
            File keyWordsArabic=new File("proj.txt");
```

```
            FileWriter writerArabic=new FileWriter(keyWordsArabic);

            testString.setText(Arabicfile);

            testString.write(writerArabic);

           String myfile="proj.txt";

           String afterstem="afterstem.txt";

           ArabicStemmer arabicStemmer = new ArabicStemmer(
myfile,afterstem);

           myfile=ChangeFiletoString(afterstem);

           String[] ArabicwordsTostemin=myfile.split("");

//  Calculate number of occurrence of each word and store them from the most
occurrence    //

int[] numOfEachWord=new int[numOfwords];

int          n=getEachWordNum(ArabicwordsTostemin,      myfileInwords,
numOfEachWord,numOfwords);

           String btext=arrange(myfileInwords,numOfEachWord,n);

           // display result into program window   //

           Tafterstem.setText(btext);

           n=getEachWordNum(myfileWithoutStop,          myfileInwords,
numOfEachWord,numOfwords);

            btext=arrange(myfileInwords,numOfEachWord,n);

           Tbeforstem.setText(btext);

           // display result into files to reuse it in the next step     //

         File keyWords=new File(wordsfile);
```

```
            FileWriter writer=new FileWriter(keyWords);
            File keyCount=new File(countfile);
            FileWriter Intwriter=new FileWriter(keyCount);
            String words="";
            String count="";
if(n>٧٠)
for(int i=٠;i<٧٠;i++)
            {
             words+=myfileInwords[i]+"\n";
             count+="١"+"\n";
            }
else
for(int i=٠;i<n;i++)
{
             words+=myfileInwords[i]+"\n";
             count+=١+"\n";
 }
           testString.setText(words);
           testInt.setText(count);
           testString.write(writer);
           testInt.write(Intwriter);
```

```
            writer.close();
             }catch(Exception ee){
             System.err.println("Error"+ ee.getMessage());
            }
    }
 ////////////////////////////////////////////////////////////
public String ChangeFiletoString(String fileName)
     {
       FileReader reader = null;
       String fileContent="";
       String test="";
try{
            File myfile         =new File(fileName);
            reader              = new FileReader(myfile);
            BufferedReader Breader=new BufferedReader(reader);
            while((test=Breader.readLine())!=null)
               fileContent+=test+" ";
            reader.close();
            Breader.close();
       } catch (Exception e) {
            System.out.println("Error"+ e.getMessage());
```

```
        }
return fileContent;
    }
   ////////////////////////////////////////////////
  public                    int                    getEachWordNum(String
[]myfileWithoutStop,String[]words,int[]numOfEachWord,int num ){
        String s="";
 int n=٠;
for(int t=٠;t<num;t++)
        {  s=myfileWithoutStop[t];
int count=٠;
if(s.equals("*"))
continue;
 else
          {
for(int i=٠;i<num;i++){
if(myfileWithoutStop[i].equals("*"))
                continue;
else if(s.equals(myfileWithoutStop[i]))
             {  count++;
             myfileWithoutStop[i]="*";}
          }
```

```
            }
            words[n]=s;
          numOfEachWord[n]=count;
            n++;
           }
           return n;
      }
      ////////////////////////////////////////////
    public String arrange (String [] myfilewords,int[] wordnum,int n)
    {
          String btext="",stemp;
  int temp=٠ ,first=٠,index=٠;
for(int i=٠;i<n;i++){
             first=wordnum[i];
for(int t=i;t<n;t++ )
if(wordnum[t]>=first){
                    first=wordnum[t];
                   index=t;
                }
              stemp=myfilewords[i];
              myfilewords[i]=myfilewords[index];
```

```
        myfilewords[index]=stemp;
                temp=wordnum[i];
                wordnum[i]=wordnum[index];
                wordnum[index]=temp;
            }
            String tryString="";
for(int e=٠;e<n;e++)
            { btext+=myfilewords[e]+ " --- >"+wordnum[e]+"\n";
              tryString+=myfilewords[e]+"\n";
            }

            File keyWords=new File("key.txt");
try {
                FileWriter tryWriter=new FileWriter(keyWords);
                testString.setText(tryString);
                testString.write(tryWriter);
            } catch (IOException ex) {

Logger.getLogger(key_words_extractor.class.getName()).log(Level.SEVERE,
null, ex);
            }
```

```
return btext;
    }
// Stemming Code……ArabicStemmer……//
import java.awt.*;
import java.awt.event.*;
import java.io.*;
import java.util.*;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.*;
import javax.swing.event.*;
publicclass ArabicStemmer {
public String str="";
protected Vector staticFiles;
 protected File currentInputFilePanelFile;
protected Vector stemmedTextLists = new Vector ( );
 //-----------------------------------------------------
     ArabicStemmer (String myfile,String afterstem)
   {JTextArea result=new JTextArea();
```

```
        readInStaticFiles ( );
        File tempFile=new File(myfile);
        currentInputFilePanelFile = tempFile;
    Stem stemmedText = new Stem ( currentInputFilePanelFile, staticFiles );
                    str=stemmedText.displayText ( );
                    result.setText(str);
                    File keyWords=new File(afterstem);
try {
            FileWriter writer=new FileWriter(keyWords);
            result.write(writer);
        } catch (IOException ex) {

Logger.getLogger(ArabicStemmer.class.getName()).log(Level.SEVERE,    null,
ex);
        }
    }
protected void readInStaticFiles ( )
    {
// create a string buffer containing the path to the static files
```

```
String pathToStemmerFiles = new StringBuffer ( System.getProperty ( "user.dir" ) + System.getProperty ( "file.separator" ) + "StemmerFiles" + System.getProperty ( "file.separator" ) ).toString ( );

// create the vector composed of vectors containing the static files

staticFiles = new Vector ( );

if ( addVectorFromFile ( new StringBuffer ( pathToStemmerFiles + "definite_article.txt" ).toString ( ) ) )

if ( addVectorFromFile ( new StringBuffer ( pathToStemmerFiles + "duplicate.txt" ).toString ( ) ) )

if ( addVectorFromFile ( new StringBuffer ( pathToStemmerFiles + "first_waw.txt" ).toString ( ) ) )

 if ( addVectorFromFile ( new StringBuffer ( pathToStemmerFiles + "first_yah.txt" ).toString ( ) ) )

if ( addVectorFromFile ( new StringBuffer ( pathToStemmerFiles + "last_alif.txt" ).toString ( ) ) )

 if ( addVectorFromFile ( new StringBuffer ( pathToStemmerFiles + "last_hamza.txt" ).toString ( ) ) )

if ( addVectorFromFile ( new StringBuffer ( pathToStemmerFiles + "last_maksoura.txt" ).toString ( ) ) )

if ( addVectorFromFile ( new StringBuffer ( pathToStemmerFiles + "last_yah.txt" ).toString ( ) ) )

 if ( addVectorFromFile ( new StringBuffer ( pathToStemmerFiles + "mid_waw.txt" ).toString ( ) ) )

    if ( addVectorFromFile ( new StringBuffer ( pathToStemmerFiles + "mid_yah.txt" ).toString ( ) ) )
```

```
if ( addVectorFromFile ( new StringBuffer ( pathToStemmerFiles +
"prefixes.txt" ).toString ( ) ) )

 if ( addVectorFromFile ( new StringBuffer ( pathToStemmerFiles +
"punctuation.txt" ).toString ( ) ) )

if ( addVectorFromFile ( new StringBuffer ( pathToStemmerFiles +
"quad_roots.txt" ).toString ( ) ) )

if ( addVectorFromFile ( new StringBuffer ( pathToStemmerFiles +
"stopwords.txt" ).toString ( ) ) )

if ( addVectorFromFile ( new StringBuffer ( pathToStemmerFiles +
"suffixes.txt" ).toString ( ) ) )

 if ( addVectorFromFile ( new StringBuffer ( pathToStemmerFiles +
"tri_patt.txt" ).toString ( ) ) )

if ( addVectorFromFile ( new StringBuffer ( pathToStemmerFiles +
"tri_roots.txt" ).toString ( ) ) )

if ( addVectorFromFile ( new StringBuffer ( pathToStemmerFiles +
"diacritics.txt" ).toString ( ) ) )

 if ( addVectorFromFile ( new StringBuffer ( pathToStemmerFiles +
"strange.txt" ).toString ( ) ) )

      {}

   }


   //---------------------------------------------------------------------

protected boolean addVectorFromFile ( String fileName )

   {
```

```
boolean returnValue;
try
        {
// the vector we are going to fill
            Vector vectorFromFile = new Vector ( );


            // create a buffered reader
            File file = new File ( fileName );
            FileInputStream fileInputStream = new FileInputStream ( file );
            InputStreamReader inputStreamReader = new InputStreamReader ( fileInputStream, "UTF-١٦" );
  //If the bufferedReader is not big enough for a file, I should change the size of it here
           BufferedReader bufferedReader = new BufferedReader ( inputStreamReader, ٢٠٠٠٠ );
// read in the text a line at a time
            String part;
            StringBuffer word = new StringBuffer ( );
while ( ( part = bufferedReader.readLine ( ) ) != null )
            {
```

```
// add spaces at the end of the line
                part = part + "  ";


                // for each line
 for ( int index = ٠; index < part.length ( ) - ١; index ++ )
                {
                   // if the character is not a space, append it to a word
if ( ! ( Character.isWhitespace ( part.charAt ( index ) ) ) )
                   {
                      word.append ( part.charAt ( index ) );
                   }
// otherwise, if the word contains some characters, add it
                   // to the vector
else
                   {
if ( word.length ( ) != ٠ )
                      {
                         vectorFromFile.addElement ( word.toString ( ) );
                         word.setLength ( ٠ );}}}}
```

```
            // destroy the buffered reader
              bufferedReader.close ( );
             fileInputStream.close ( );
            staticFiles.addElement ( vectorFromFile );
            returnValue = true;
        }
catch ( Exception exception )
        { returnValue = false;
        }
return returnValue;
    }
}

//……….stem.java……..//
import java.io.*;
import java.util.*;
import java.text.DateFormat;
```

```
/**
 *
 * @author Owners
 */
public class Stem {
// one line of text
 private String oneLine;
 // all text to be passed to the text window
 private StringBuffer all = new StringBuffer( "" );
// the stemmed Arabic text
private Vector stemmedText = new Vector( );

   // the tokenized line with all punctuation removed
private Vector tokenizedLine = new Vector( );

// the files containing prefixes, suffixes and so on
private Vector staticFiles;

// have the root, pattern, stopword or strange words been found
```

```
private boolean rootFound = false;
private boolean patternFound = false;
private boolean stopwordFound = false;
private boolean strangeWordFound = false;
private boolean rootNotFound = false;
private boolean fromSuffixes = false;
 private String[] stemmedDocument;
private int wordNumber          = ٠;
private int notWordNumber       = ٠;
private Vector listStemmedWords = new Vector( );
private Vector listNotStemmedWords = new Vector( );
private Vector listStopwordsFound = new Vector( );
 private Vector listOriginalStopword = new Vector( );
 private Vector listsVector = new Vector( );
 private Vector wordsNotStemmed = new Vector( );
 private Vector wordsWithNoRoots = new Vector( );
 int number = ٠;

//-----------------------------------------------------------------------
```

```
 // constructor
 public Stem ( File fileToBeStemmed, Vector statFiles )
    {
// clone the static files
        staticFiles = ( Vector ) statFiles.clone ( );

// create the statistics file array
        stemmedDocument = new String[١٠٠٠];
        // read the contents of the file, one line at a time
        // after each line stem the words
        readFromFile ( fileToBeStemmed );
for ( int i=٠; i<wordNumber; i++ )
            all.append ( stemmedDocument[i] +"\n");
    }

//-----------------------------------------------------------------------

    // return the results
public String displayText ( )
    {
```

```
 return ( all.toString ( ) );
    }
     // read the contents of the file and tokenize the text
private void readFromFile ( File currentFile )
    {
try {
// read from the file using FileReader
            FileInputStream in = new FileInputStream ( currentFile );
            InputStreamReader str = new InputStreamReader ( in, "UTF-٨" );
            BufferedReader br = new BufferedReader ( str );
// initialize index
            StringBuffer word = new StringBuffer ( );
            Character character;
            String currentWord;
int lineNumber = ١;

// read in the text a line at a time
 while ( ( oneLine = br.readLine ( ) ) != null )
            {
```

```
            // add spaces at the end of the line
            oneLine = oneLine + "  ";
            lineNumber++;
// tokenize each line
for ( int i=٠; i<oneLine.length ( ); i++ )
            {
               // if the character is not a space, append it to a word
if( ( ! ( character = new Character ( oneLine.charAt ( i ) ) ).isWhitespace ( oneLine.charAt ( i ) ) ) )
               {
                  word.append ( oneLine.charAt ( i ) );
               }
               // otherwise, if the word contains some characters, add it to the vector
               else
               {
 if ( word.length ( ) != ٠ )
                  {
                     tokenizedLine.addElement ( word.toString ( ) );
                     word.setLength ( ٠ );
                  }
               }
```

```
            }
    // now we have tokenized one line, we should stem it
    for ( int i = ٠; i < tokenizedLine.size ( ); i++ )
                {
                    // set the word in a string
                    currentWord = tokenizedLine.elementAt ( i ) .toString ( );

    // store the original word in the array stemmedDocument
     stemmedDocument[wordNumber] = currentWord;

                    // stem the word
                    currentWord = formatWord ( currentWord, i );

    // if the word wasn't stemmed, indicate this in stemmedDocument
     if ( stemmedDocument[wordNumber] == null )
                    {

stemmedDocument[wordNumber]= currentWord;
 listNotStemmedWords.addElement ( currentWord );
                }
```

```
                wordNumber++;
// re-initialize the variable rootFound
rootFound = false;
// add the stemmed word to the vector
                  stemmedText.addElement ( currentWord );
               }

// after adding all the stemmed word on this line, we should add a new line character
               stemmedText.addElement ( "\n" );
// after we have finished processing this line we should clear it
               tokenizedLine.removeAllElements ( );

            }
           // close the FileReader
           br.close ( );
                in.close ( );
           }
            catch ( IOException e )
                  {

                  }
```

```
    }


  //---------------------------------------------------------------------


  // format the word by removing any punctuation, diacritics and non-letter
charracters

  private String formatWord ( String currentWord, int index )

    {

        StringBuffer modifiedWord = new StringBuffer ( );


  // remove any diacritics (short vowels)




if ( removeDiacritics( currentWord, modifiedWord ) )

        {

            tokenizedLine.setElementAt ( currentWord = modifiedWord.toString (
), index );

        }


       // remove any punctuation from the word

if ( removePunctuation( currentWord, modifiedWord ) )
```

```
        {
            tokenizedLine.setElementAt ( currentWord = modifiedWord.toString (
), index );
        }

 // there could also be characters that aren't letters which should be removed
        if ( removeNonLetter ( currentWord, modifiedWord ) )
        {
            tokenizedLine.setElementAt ( currentWord = modifiedWord.toString (
), index );
        }

// check for stopwords


if( !checkStrangeWords ( currentWord ) )
// check for stopwords
if( !checkStopwords ( currentWord ) )
                currentWord = stemWord ( currentWord );
 return currentWord;
    }
//---------------------------------------------------------------------
```

```
    // stem the word
 private String stemWord ( String word )
    {
        // check if the word consists of two letters
// and find it's root
        if ( word.length ( ) == ٢ )
            word = isTwoLetters ( word );
// if the word consists of three letters
if( word.length ( ) == ٣ && !rootFound )
// check if it's a root
            word = isThreeLetters ( word );
        // if the word consists of four letter

if( word.length ( ) == ٤ )
            // check if it's a root
            isFourLetters ( word );
 // if the root hasn't yet been found
if( !rootFound )
        {
            // check if the word is a pattern
```

```
            word = checkPatterns ( word );
        }

        // if the root still hasn't been found
        if ( !rootFound )
        {
// check for a definite article, and remove it
            word = checkDefiniteArticle ( word );
        }
// if the root still hasn't been found
 if ( !rootFound && !stopwordFound )
        {


// check for the prefix waw
            word = checkPrefixWaw ( word );
        }
        // if the root STILL hasnt' been found
        if ( !rootFound && !stopwordFound )
        {
            // check for suffixes
```

```
            word = checkForSuffixes ( word );
        }

// if the root STILL hasn't been found
if ( !rootFound && !stopwordFound )
        {
            // check for prefixes
            word = checkForPrefixes ( word );
        }

 return word;
    }


//------------------------------------------------------------------------

    // check and remove any prefixes
private String checkForPrefixes ( String word )
    {
        String prefix = "";
        String modifiedWord = word;
```

```
        Vector prefixes = ( Vector ) staticFiles.elementAt ( ١٠ );

// for every prefix in the list
for ( int i = ٠; i < prefixes.size ( ); i++ )
        {
           prefix = ( String ) prefixes.elementAt ( i );
// if the prefix was found
if ( prefix.regionMatches ( ٠, modifiedWord, ٠, prefix.length ( ) ) )
           {
              modifiedWord = modifiedWord.substring ( prefix.length ( ) );

// check to see if the word is a stopword
              if ( checkStopwords( modifiedWord ) )
 return modifiedWord;

              // check to see if the word is a root of three or four letters
              // if the word has only two letters, test to see if one was removed
              if ( modifiedWord.length ( ) == ٢ )
                 modifiedWord = isTwoLetters ( modifiedWord );
else if ( modifiedWord.length ( ) == ٣ && !rootFound )
                 modifiedWord = isThreeLetters ( modifiedWord );
```

```
else if ( modifiedWord.length ( ) == ٤ )
                isFourLetters ( modifiedWord );
// if the root hasn't been found, check for patterns
 if ( !rootFound && modifiedWord.length ( ) > ٢ )
                modifiedWord = checkPatterns ( modifiedWord );

  // if the root STILL hasn't been found
if ( !rootFound && !stopwordFound && !fromSuffixes)
                {
// check for suffixes
                modifiedWord = checkForSuffixes ( modifiedWord );
                }

if ( stopwordFound )
return modifiedWord;
// if the root was found, return the modified word
if ( rootFound && !stopwordFound )
                {
return modifiedWord;
                }
```

```
            }
        }
return word;
    }
    //---------------------------------------------------------------------
// METHOD CHECKFORSUFFIXES
 private String checkForSuffixes ( String word )
    {
        String suffix = "";
        String modifiedWord = word;
        Vector suffixes = ( Vector ) staticFiles.elementAt ( ١٤ );
        fromSuffixes = true;

// for every suffix in the list
 for ( int i = ٠; i < suffixes.size ( ); i++ )
        {
            suffix = ( String ) suffixes.elementAt ( i );
// if the suffix was found
if( suffix.regionMatches ( ٠, modifiedWord, modifiedWord.length ( ) -
suffix.length ( ), suffix.length ( ) ) )
            {
```

```
            modifiedWord = modifiedWord.substring ( ٠, modifiedWord.length (
) - suffix.length ( ) );


// check to see if the word is a stopword
if ( checkStopwords ( modifiedWord ) )
            {
               fromSuffixes = false;
return modifiedWord;
            }


// check to see if the word is a root of three or four letters
            // if the word has only two letters, test to see if one was removed
if ( modifiedWord.length ( ) == ٢ )
            {
               modifiedWord = isTwoLetters ( modifiedWord );
            }
else if ( modifiedWord.length ( ) == ٣ )
            {
               modifiedWord = isThreeLetters ( modifiedWord );
            }
else if ( modifiedWord.length ( ) == ٤ )
            {
```

```
            isFourLetters ( modifiedWord );
        }

// if the root hasn't been found, check for patterns
        if ( !rootFound && modifiedWord.length( ) > ٢ )
        {
            modifiedWord = checkPatterns( modifiedWord );
        }

if ( stopwordFound )
        {
            fromSuffixes = false;

 return modifiedWord;
        }
    // if the root was found, return the modified word
        if ( rootFound )
        {
            fromSuffixes = false;
return modifiedWord;
        }
```

```
            }
        }
        fromSuffixes = false;
return word;
    }

//---------------------------------------------------------------------
 // check and remove the special prefix (waw)
private String checkPrefixWaw ( String word )
    {
        String modifiedWord = "";
if ( word.length ( ) > ٣ && word.charAt ( ٠ ) == '\u٠٦٤٨' )
        {
            modifiedWord = word.substring ( ١ );

            // check to see if the word is a stopword
            if ( checkStopwords ( modifiedWord ) )
 return modifiedWord;

            // check to see if the word is a root of three or four letters
            // if the word has only two letters, test to see if one was removed
```

```
 if ( modifiedWord.length ( ) == ٢ )
            modifiedWord = isTwoLetters( modifiedWord );
         else if ( modifiedWord.length ( ) == ٣ && !rootFound )
            modifiedWord = isThreeLetters( modifiedWord );
         else if ( modifiedWord.length ( ) == ٤ )
            isFourLetters ( modifiedWord );

// if the root hasn't been found, check for patterns
if ( !rootFound && modifiedWord.length ( ) > ٢ )
            modifiedWord = checkPatterns ( modifiedWord );

// if the root STILL hasnt' been found
 if ( !rootFound && !stopwordFound )
         {
            // check for suffixes
            modifiedWord = checkForSuffixes ( modifiedWord );
         }

         // iIf the root STILL hasn't been found
if ( !rootFound && !stopwordFound )
         {
```

```
// check for prefixes
                modifiedWord = checkForPrefixes ( modifiedWord );
            }

if ( stopwordFound )
return modifiedWord;
if ( rootFound && !stopwordFound )
            {
return modifiedWord;



            }
        }
 return word;
    }
//---------------------------------------------------------------------
   // check and remove the definite article
 private String checkDefiniteArticle ( String word )
    {
// looking through the vector of definite articles
        // search through each definite article, and try and
```

```
        // find a match
        String definiteArticle = "";
        String modifiedWord = "";
        Vector definiteArticles = ( Vector ) staticFiles.elementAt ( ٠ );
// for every definite article in the list
for ( int i = ٠; i < definiteArticles.size ( ); i++ )
        {
           definiteArticle = ( String ) definiteArticles.elementAt ( i );
           // if the definite article was found
if ( definiteArticle.regionMatches ( ٠, word, ٠, definiteArticle.length ( ) ) )
           {
              // remove the definite article
              modifiedWord = word.substring ( definiteArticle.length ( ),
word.length ( ) );

              // check to see if the word is a stopword
 if ( checkStopwords ( modifiedWord ) )
return modifiedWord;

              // check to see if the word is a root of three or four letters
              // if the word has only two letters, test to see if one was removed
 if ( modifiedWord.length ( ) == ٢ )
```

```
                modifiedWord = isTwoLetters ( modifiedWord );
else if ( modifiedWord.length ( ) == ٣ && !rootFound )
                modifiedWord = isThreeLetters ( modifiedWord );
else if ( modifiedWord.length ( ) == ٤ )
                isFourLetters ( modifiedWord );
            // if the root hasn't been found, check for patterns
if ( !rootFound && modifiedWord.length ( ) > ٢ )
                modifiedWord = checkPatterns ( modifiedWord );


// if the root STILL hasnt' been found
if ( !rootFound&& !stopwordFound )
            {
// check for suffixes
                modifiedWord = checkForSuffixes ( modifiedWord );
            }
// if the root STILL hasn't been found
 if ( !rootFound && !stopwordFound )
            {
// check for prefixes
                modifiedWord = checkForPrefixes ( modifiedWord );
            }
```

```
 if ( stopwordFound )
return modifiedWord;

 // if the root was found, return the modified word
if ( rootFound && !stopwordFound )
            {

return modifiedWord;
            }
          }
       }
 if ( modifiedWord.length ( ) > ٣ )
return modifiedWord;
return word;
   }

//---------------------------------------------------------------------
   // if the word consists of two letters
 private String isTwoLetters ( String word )
   {
// if the word consists of two letters, then this could be either
```

```
        // - because it is a root consisting of two letters (though I can't think of any!)
        // - because a letter was deleted as it is duplicated or a weak middle or last letter.
        word = duplicate ( word );
// check if the last letter was weak

 if ( !rootFound )
            word = lastWeak ( word );
// check if the first letter was weak
if ( !rootFound )
            word = firstWeak ( word );

        // check if the middle letter was weak
 if ( !rootFound )
            word = middleWeak ( word );

 return word;
    }
//-----------------------------------------------------------------------
    // if the word consists of three letters
    private String isThreeLetters ( String word )
```

```
    {
        StringBuffer modifiedWord = new StringBuffer ( word );
        String root = "";
        // if the first letter is a '?', '?'  or '?'
        // then change it to a '?'
if ( word.length ( ) > ٠ )
        {
          if ( word.charAt ( ٠ ) == '\u٠٦٢٧' || word.charAt ( ٠ ) == '\u٠٦٢٤' ||
word.charAt ( ٠ ) == '\u٠٦٢٦' )
          {
             modifiedWord.setLength ( ٠ );
             modifiedWord.append ( '\u٠٦٢٣' );
             modifiedWord.append ( word.substring ( ١ ) );
             root = modifiedWord.toString ( );
          }
// if the last letter is a weak letter or a hamza
          // then remove it and check for last weak letters
          if ( word.charAt ( ٢ ) == '\u٠٦٤٨' || word.charAt ( ٢ ) == '\u٠٦٤a' ||
word.charAt ( ٢ ) == '\u٠٦٢٧' ||
             word.charAt ( ٢ ) == '\u٠٦٤٩' || word.charAt ( ٢ ) == '\u٠٦٢١' ||
word.charAt ( ٢ ) == '\u٠٦٢٦' )
          {
             root = word.substring ( ٠, ٢ );
```

```
            root = lastWeak ( root );
            if ( rootFound )
            {
                return root;
            }
        }
        // if the second letter is a weak letter or a hamza
        // then remove it
if ( word.charAt ( ١ ) == '\u٠٦٤٨' || word.charAt ( ١ ) == '\u٠٦٤a' || word.charAt
( ١ ) == '\u٠٦٢٧' || word.charAt ( ١ ) == '\u٠٦٢٦' )
        {
            root = word.substring ( ٠, ١ );
            root = root + word.substring ( ٢ );
            root = middleWeak ( root );
if ( rootFound )
            {
 return root;
            }
        }
        // if the second letter has a hamza, and it's not on a alif
```

```
        // then it must be returned to the alif
        if ( word.charAt ( ١ ) == '\u٠٦٢٤' || word.charAt ( ١ ) == '\u٠٦٢٦' )
        {
 if ( word.charAt ( ٢ ) == '\u٠٦٤٥' || word.charAt ( ٢ ) == '\u٠٦٣٢' ||
word.charAt ( ٢ ) == '\u٠٦٣١' )
            {
               root = word.substring ( ٠, ١ );
               root = root + '\u٠٦٢٧';
               root = root+ word.substring ( ٢ );
            }
            else
            {
               root = word.substring ( ٠, ١ );
               root = root + '\u٠٦٢٣';
               root = root + word.substring ( ٢ );
            }
        }
// if the last letter is a shadda, remove it and
        // duplicate the last letter
 if ( word.charAt ( ٢ ) == '\u٠٦٥١')
        {
```

```
                root = word.substring ( ٠, ١ );
                root = root + word.substring ( ١, ٢ );
            }
        }

// if word is a root, then rootFound is true
 if ( root.length ( ) == ٠ )
        {
if ( ( ( Vector ) staticFiles.elementAt ( ١٦ ) ) .contains ( word ) )
            {
                rootFound = true;
                stemmedDocument[wordNumber] = word;
                listStemmedWords.addElement   (   stemmedDocument[wordNumber]
);
 if ( rootNotFound )
                {
for ( int i = ٠; i < number; i++ )
 wordsNotStemmed.removeElement( wordsNotStemmed.lastElement ( ) );
rootNotFound = false;
                }
 return word;
            }
```

```
        }
// check for the root that we just derived
        else if ( ( ( Vector ) staticFiles.elementAt ( ١٦ ) ) .contains ( root ) )
        {
rootFound = true;
stemmedDocument[wordNumber] = root;
 listStemmedWords.addElement ( stemmedDocument[wordNumber] );
if ( rootNotFound )
            {
               for ( int i = ٠; i < number; i++ )
wordsNotStemmed.removeElement ( wordsNotStemmed.lastElement ( ) );
rootNotFound = false;
            }
return root;
        }

 return word;
    }
```

```
   // if the word has four letters
privatevoid isFourLetters ( String word )
   {
// if word is a root, then rootFound is true
if( ( ( Vector ) staticFiles.elementAt ( ١٢ ) ) .contains ( word ) )
      {
         rootFound = true;


         stemmedDocument[wordNumber] = word;
         listStemmedWords.addElement ( stemmedDocument[wordNumber] );


      }
   }
 //---------------------------------------------------------------------
   // check if the word matches any of the patterns
 private String checkPatterns ( String word )
   {
      StringBuffer root = new StringBuffer ( "" );
      // if the first letter is a hamza, change it to an alif
```

```
if ( word.length ( ) > ٠ )

 if ( word.charAt ( ٠ ) == '\u٠٦٢٣' || word.charAt ( ٠ ) == '\u٠٦٢٥' ||
word.charAt ( ٠ ) == '\u٠٦٢٢' )

        {

           root.append ( "j" );

           root.setCharAt ( ٠, '\u٠٦٢٧' );

           root.append ( word.substring ( ١ ) );

           word = root.toString ( );

        }
// try and find a pattern that matches the word

      Vector patterns = ( Vector ) staticFiles.elementAt ( ١٥ );

      int numberSameLetters = ٠;

      String pattern = "";

      String modifiedWord = "";
// for every pattern

for( int i = ٠; i < patterns.size ( ); i++ )

      {

         pattern = ( String ) patterns.elementAt ( i );

         root.setLength ( ٠ );
// if the length of the words are the same
```

```
if ( pattern.length ( ) == word.length ( ) )
        {
          numberSameLetters = ٠;
          // find out how many letters are the same at the same index
          // so long as they're not a fa, ain, or lam
for ( int j = ٠; j < word.length ( ); j++ )
if ( pattern.charAt ( j ) == word.charAt ( j ) &&
                pattern.charAt ( j ) != '\u٠٦٤١'&&
                pattern.charAt ( j ) != '\u٠٦٣٩'&&
                pattern.charAt ( j ) != '\u٠٦٤٤'          )
               numberSameLetters ++;
// test to see if the word matches the pattern
 if ( word.length ( ) == ٦ && word.charAt ( ٣ ) == word.charAt ( ٥ ) &&
numberSameLetters == ٢ )
          {
            root.append ( word.charAt ( ١ ) );
            root.append ( word.charAt ( ٢ ) );
            root.append ( word.charAt ( ٣ ) );
            modifiedWord = root.toString ( );
```

```
    modifiedWord = isThreeLetters ( modifiedWord );
     if ( rootFound )
     return modifiedWord;
                    else
                       root.setLength ( ٠ );
                  }
// if the word matches the pattern, get the root
      if ( word.length ( ) - ٣ <= numberSameLetters )
      {
     // derive the root from the word by matching it with the pattern
     for ( int j = ٠; j < word.length ( ); j++ )
      if ( pattern.charAt ( j ) == '\u٠٦٤١' ||
                          pattern.charAt ( j ) == '\u٠٦٣٩' ||
                          pattern.charAt ( j ) == '\u٠٦٤٤'  )
                         root.append ( word.charAt ( j ) );
                    modifiedWord = root.toString ( );
                    modifiedWord = isThreeLetters ( modifiedWord );
                    if ( rootFound )
                    {
                       word = modifiedWord;
                       return word;
```

```
                }
            }
        }
    }
return word;
  }
//---------------------------------------------------------------------
 // remove non-letters from the word
private boolean removeNonLetter ( String currentWord, StringBuffer modifiedWord )
  {
    boolean nonLetterFound = false;
    modifiedWord.setLength ( ٠ );
 // if any of the word is not a letter then remove it
 for( int i = ٠; i < currentWord.length ( ); i++ )
    {
if ( Character.isLetter ( currentWord.charAt ( i ) ) )
        {
          modifiedWord.append ( currentWord.charAt ( i ) );
```

```
stemmedDocument[wordNumber] = null;
            }
            else
            {
                nonLetterFound = true;
 if ( modifiedWord.length ( ) == ٠ &&stemmedDocument[wordNumber] ==
null )
                {
notWordNumber ++;
                }
            }
        }
 return nonLetterFound;
    }
//---------------------------------------------------------------------
// handle duplicate letters in the word
private String duplicate ( String word )
    {
// check if a letter was duplicated
 if ( ( ( Vector ) staticFiles.elementAt ( ١ ) ).contains ( word ) )
```

```
        {
// if so, then return the deleted duplicate letter
            word = word + word.substring ( ١ );

// root was found, so set variable
rootFound = true;
stemmedDocument[wordNumber] = word;
listStemmedWords.addElement ( stemmedDocument[wordNumber] );
 return word;
        }
return word;
    }

//---------------------------------------------------------------------
    // check if the last letter of the word is a weak letter
private String lastWeak ( String word )
    {
        StringBuffer stemmedWord = new StringBuffer ( "" );
```

```
// check if the last letter was an alif
if ( ( ( Vector )staticFiles.elementAt ( ٤ ) ).contains ( word ) )
        {
           stemmedWord.append ( word );
           stemmedWord.append ( "\u٠٦٢٧" );
           word = stemmedWord.toString ( );
           stemmedWord.setLength ( ٠ );
// root was found, so set variable
           rootFound = true;
stemmedDocument[wordNumber] = word;
 listStemmedWords.addElement ( stemmedDocument[wordNumber] );
 return word;
        }
// check if the last letter was an hamza
else if ( ( ( Vector ) staticFiles.elementAt ( ٥ ) ) .contains ( word ) )
        {
           stemmedWord.append ( word );
           stemmedWord.append ( "\u٠٦٢٣" );



           word = stemmedWord.toString ( );
```

```
            stemmedWord.setLength ( ٠ );
// root was found, so set variable
            rootFound = true;
            stemmedDocument[wordNumber] = word;
listStemmedWords.addElement ( stemmedDocument[wordNumber] );

 return word;
        }
// check if the last letter was an maksoura
else if ( ( ( Vector ) staticFiles.elementAt ( ٦ ) ) .contains ( word ) )
        {
            stemmedWord.append ( word );
            stemmedWord.append ( "\u٠٦٤٩" );
            word = stemmedWord.toString ( );
stemmedWord.setLength ( ٠ );
// root was found, so set variable
 rootFound = true;
            stemmedDocument[wordNumber]= word;
 listStemmedWords.addElement ( stemmedDocument[wordNumber]);

 return word;
```

```
        }


// check if the last letter was an yah
else if ( ( ( Vector ) staticFiles.elementAt ( ٧ ) ).contains ( word ) )
        {
            stemmedWord.append ( word );
            stemmedWord.append ( "\u٠٦٤a" );
            word = stemmedWord.toString ( );
            stemmedWord.setLength ( ٠ );
// root was found, so set variable
rootFound = true;
stemmedDocument[wordNumber]= word;
 listStemmedWords.addElement ( stemmedDocument[wordNumber]);
return word;
        }
 return word;
    }


//---------------------------------------------------------------------
```

```
// check if the first letter is a weak letter
 private String firstWeak ( String word )
   {
      StringBuffer stemmedWord = new StringBuffer ( "" );
 // check if the firs letter was a waw
if( ( ( Vector ) staticFiles.elementAt ( ٢ ) ) .contains ( word ) )
      {
         stemmedWord.append ( "\u٠٦٤٨" );
         stemmedWord.append ( word );
         word = stemmedWord.toString ( );
         stemmedWord.setLength ( ٠ );
         // root was found, so set variable
rootFound = true;
stemmedDocument[wordNumber]= word;
 listStemmedWords.addElement ( stemmedDocument[wordNumber]);
 return word;
      }
      // check if the first letter was a yah


else if ( ( ( Vector ) staticFiles.elementAt ( ٣ ) ) .contains ( word ) )
```

```
        {
            stemmedWord.append ( "\u٠٦٤a" );
            stemmedWord.append ( word );
            word = stemmedWord.toString ( );
            stemmedWord.setLength ( ٠ );
// root was found, so set variable
 rootFound = true;
stemmedDocument[wordNumber] = word;
listStemmedWords.addElement ( stemmedDocument[wordNumber]);
 return word;
        }
 return word;
    }
//---------------------------------------------------------------------
// check if the middle letter of the root is weak
 private String middleWeak ( String word )
    {
        StringBuffer stemmedWord = new StringBuffer ( "j" );


        // check if the middle letter is a waw
```

```
 if ( ( ( Vector ) staticFiles.elementAt ( ٨ ) ) .contains ( word ) )
        {
// return the waw to the word
            stemmedWord.setCharAt ( ٠, word.charAt ( ٠ ) );
            stemmedWord.append ( "\u٠٦٤٨" );
            stemmedWord.append ( word.substring ( ١ ) );
            word = stemmedWord.toString ( );
            stemmedWord.setLength ( ٠ );
            // root was found, so set variable
            rootFound = true;
 stemmedDocument[wordNumber] = word;
listStemmedWords.addElement ( stemmedDocument[wordNumber] );
            return word;
        }
// check if the middle letter is a yah
else if ( ( ( Vector ) staticFiles.elementAt ( ٩ ) ) .contains ( word ) )
        {
            // return the waw to the word



            stemmedWord.setCharAt ( ٠, word.charAt ( ٠ ) );
```

```
            stemmedWord.append ( "\u٠٦٤a" );
            stemmedWord.append ( word.substring ( ١ ) );
            word = stemmedWord.toString ( );
            stemmedWord.setLength ( ٠ );
            // root was found, so set variable//
rootFound = true;
stemmedDocument[wordNumber]= word;
 listStemmedWords.addElement ( stemmedDocument[wordNumber] );
 return word;
        }
return word;
    }
//---------------------------------------------------------------------
  // remove punctuation from the word
private boolean removePunctuation ( String currentWord, StringBuffer
modifiedWord )
    {
 boolean punctuationFound = false;
        modifiedWord.setLength ( ٠ );


        Vector punctuations = ( ١          ١٢٧        es.elementAt ( ١١ );
```

```
// for every character in the current word, if it is a punctuation then do nothing
        // otherwise, copy this character to the modified word
 for ( int i = ٠; i < currentWord.length ( ); i++ )
        {
 if ( ! ( punctuations.contains ( currentWord.substring ( i, i+١ ) ) ) )
            {
                modifiedWord.append ( currentWord.charAt ( i ) );
stemmedDocument[wordNumber] = null;
            }

        }
 return punctuationFound;
    }
    //---------------------------------------------------------------------
// remove diacritics from the word//
private boolean removeDiacritics ( String currentWord, StringBuffer
modifiedWord )
    {

boolean diacriticFound = false;
                                                    ١٢٨
        modifiedWord.setLength ( ٠ ),
```

```
        Vector diacritics = ( Vector ) staticFiles.elementAt ( ١٧ );
        for ( int i = ٠; i < currentWord.length ( ); i++ )
           // if the character is not a diacritic, append it to modified word
if ( ! ( diacritics.contains ( currentWord.substring ( i, i+١ ) ) ) )
              modifiedWord.append ( currentWord.substring ( i, i+١ ) );
           else
           {
              diacriticFound = true;
           }
return diacriticFound;
   }
//---------------------------------------------------------------------
// check that the word is a stopword
private boolean checkStopwords ( String currentWord )
   {
       Vector v = ( Vector ) staticFiles.elementAt ( ١٣ );
if ( stopwordFound = v.contains ( currentWord ) )
       {

stemmedDocument[wordNumber]= currentWord;
listStopwordsFound.addElement( currentWord );
```

```
 listOriginalStopword.addElement( stemmedDocument[wordNumber] );
        }
        returnstopwordFound;
    }
    //-----------------------------------------------------------------------
    // check that the word is a strange word
private boolean checkStrangeWords ( String currentWord )
    {
        Vector v = ( Vector ) staticFiles.elementAt ( ١٨ );

        if ( strangeWordFound = v.contains( currentWord ) )
        {
stemmedDocument[wordNumber] = currentWord;
listStopwordsFound.addElement( currentWord );
 listOriginalStopword.addElement( stemmedDocument[wordNumber] );
        }



        returnstrangeWordFound;
    }
```

```
}
```

Classification step, now we try to classify text to show that it belongs to political field or sportive or economic field and so on.

```
// change each file words and count to array then we have to arrays one for words and another for counter//
    String all=ChangeFiletoString (politicalword);
    String[]politics=all.split("");
    all=ChangeFiletoString(politicalcount );
    String[]healthcon=all.split("");
////////////////////////////////////
    all=ChangeFiletoString(sportword );
    String[]sport=all.split("");
    all=ChangeFiletoString(sportcount );
    String[]sportcon=all.split("");
////////////////////////////////////
     all=ChangeFiletoString(economicsword );
    String[]economics=all.split("");
    all=ChangeFiletoString(economicscount );
    String[]economicscon=all.split("");
////////////////////////////////////
    all=ChangeFiletoString(filewords);
    String[]myfile=all.split("");
    all=ChangeFiletoString(filecount);
    String[]classfilecount=all.split("");
//////////////////////////////////////////////////////
// merge file of words with file that you want to make check on it//
int num=dmg(totalpolitical,politics,myfile);
    String []file١=new String[num];
    String []file٢=new String[num];




    vector(file١, myfile, classfilecount, totalpolitical, num);
    vector(file٢, politics, healthco          ١٣١          num);
//////////////////////////////////////////////////////
    num=dmg(totalsport,sport,myfile);
```

```
      String []file٣=new String[num];
      String []file٤=new String[num];
      vector(file٣, myfile, classfilecount, totalsport, num);
      vector(file٤, sport, sportcon, totalsport, num);
      ////////////////////////////////////////////////////
      num=dmg(totaleconomics,economics,myfile);
      String []file٥=new String[num];
      String []file٦=new String[num];
      vector(file٥, myfile, classfilecount, totaleconomics, num);
      vector(file٦, economics, economicscon, totaleconomics,    num);
      ///////////////////////////////////////////////////////
double x=dis(file١,file٢);
    double y=dis(file٣,file٤);
    double z=dis(file٥,file٦);
    String it="";
if(x<y&&x<z)
     it="Health";
else if(y<x&&y<z)
     it="Sport";
else if(z<x&&z<y)
     it="Economics";
distance="Health:"+x+"\nSport:"+y+"\nEconomics: "+z+"\n\n This Topic is :"+it;
  //////////////////////////////////////////////////////////////////////////////////////////////////
            double a=cosin(file١,file٢);
            double b=cosin(file٣,file٤);
            double c=cosin(file٥,file٦);
          if(a>b&&a>c)
              it="Health";
          else if(b>a&&b>c)
              it="Sport";
          else if(c>a&&c>b)
              it="Economics";
          cosin="Health:"+a+"\nSpo[illegible]nomics:"+c+"\n\n     This   Topic
          is:"+it;

            }
```

```
//////////////////////////////////////
publicString ChangeFiletoString(String fileName)
 {
        FileReader reader ;
        String fileContent=" ";
        String test;
        try{
        File myfile=newFile(fileName);
        reader =newFileReader(myfile);
        BufferedReader Breader=newBufferedReader(reader);
        while((test=Breader.readLine())!=null)
        fileContent+=test+" ";
        reader.close();
        Breader.close();
              }catch(Exception e) {
        System.out.println("Error"+ e.getMessage());
         }
        returnfileContent;
 }
////////////////////////////////////////////////////////////////////

public int dmg(String []total,String[] file١,String[]file٢){
        for(int i=٠;i<file١.length;i++)
              total[i]=file١[i];
        int num=file١.length;
        int test=٠;
        for(int i=٠;i<file٢.length;i++)
              {      test=٠;




              for(int t=٠;t<file١.le                    ١٣٣
              if(file٢[i].equals(file
                        test++ ;
              if(test ==٠)
                        total[num++]=file٢[i];
```

```
                }
return num;

}
                ////////////////////////////////////////////////////////////////////
Publicvoid
vector(String[]result,String[]filetocheckW,String[]filetochekC,String[]filetocheck
With,int num){
int index=-١;
for(int i=٠;i<num;i++)
            {
                index=-١;
        for(int t=٠;t<filetocheckW.length;t++)
            {
        if(filetocheckWith[i].equals(filetocheckW[t]))
                index=t;
            }
        if(index!=-١)
                result[i]=filetochekC[index];
            else
             result[i]="٠";
    }
}
////////////////////////////////////////////////////////////////////////
```

Calculate distance and cosine to determine which field it belongs to.

```
        publicdouble dis(String []file١,String[] file٢){
            int t=٠,x=٠;
            for(int i=٠;i<file١.length;i++){




                        t=٠;
                        t=Integer.p          ١٣٤          -Integer.parseInt(file٢[i]);
                        t*=t;
                        x+=t;
                }
```

```
            double y=Math.sqrt(x);
            return y;
        }


////////////////////////////////////////////////////////////////////

publicdouble cosin(String[]file١,String[]file٢){
        int  x=٠,t=٠;int z=٠;double y=٠.٠;
        for(int i=٠;i<file١.length;i++){
                x+=Integer.parseInt(file١[i])*Integer.parseInt(file٢[i]);
                t+=Integer.parseInt(file١[i])*Integer.parseInt(file١[i]);
                z+=Integer.parseInt(file٢[i])*Integer.parseInt(file٢[i]);
            }

 double a=٠.٠,b=٠.٠,c=٠.٠;

    a=Math.sqrt(t);
    b=Math.sqrt(z);
    c=a*b;
    y=x/c;
 return y;

}
}
```

Now we have some result to this mining in our project.

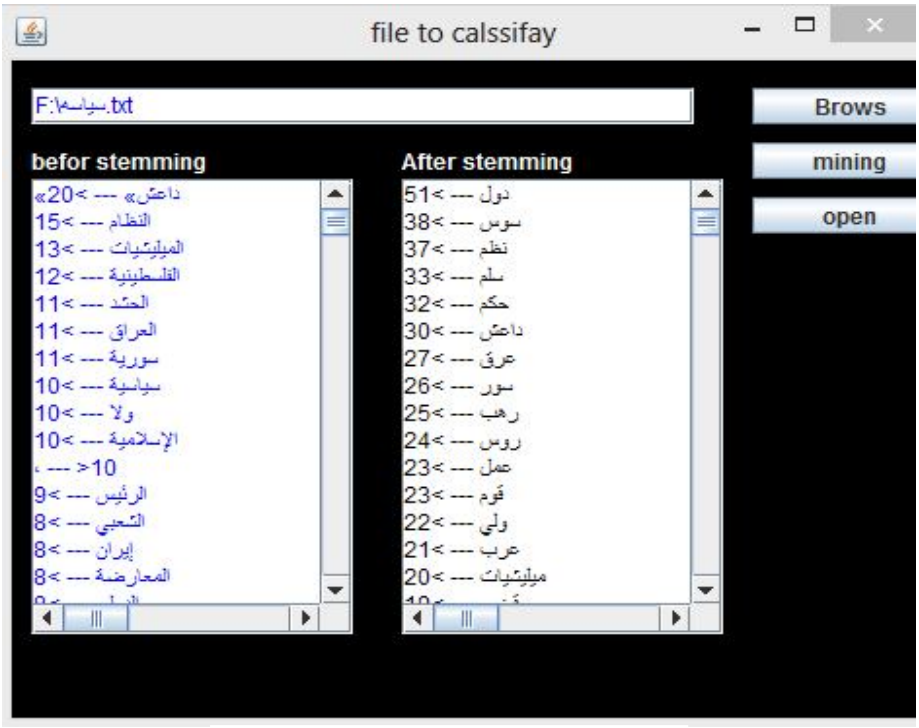


Figure ٥.٦(١)

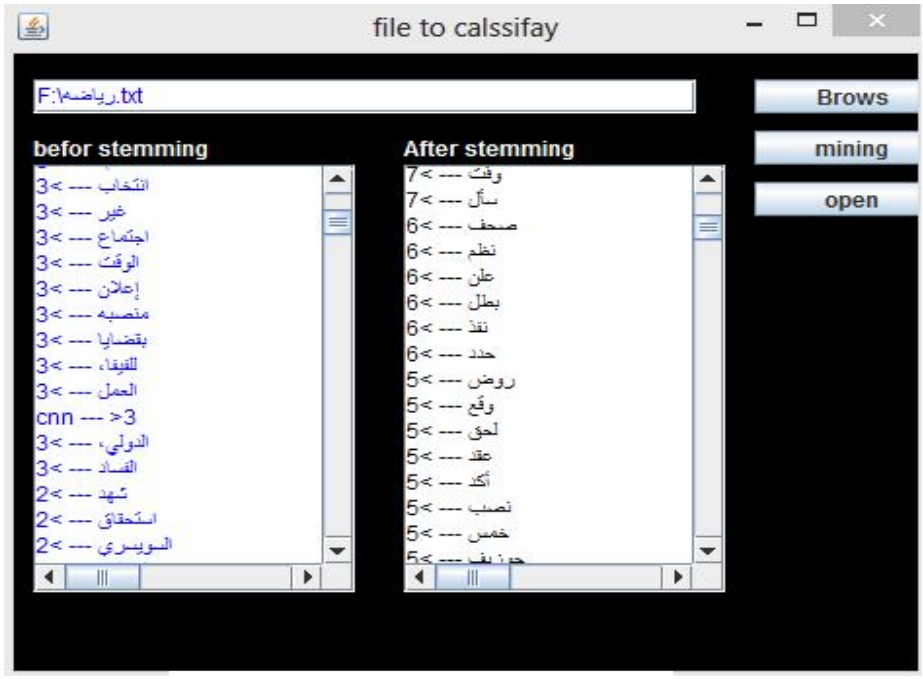


Figure ٥.٦(٢)

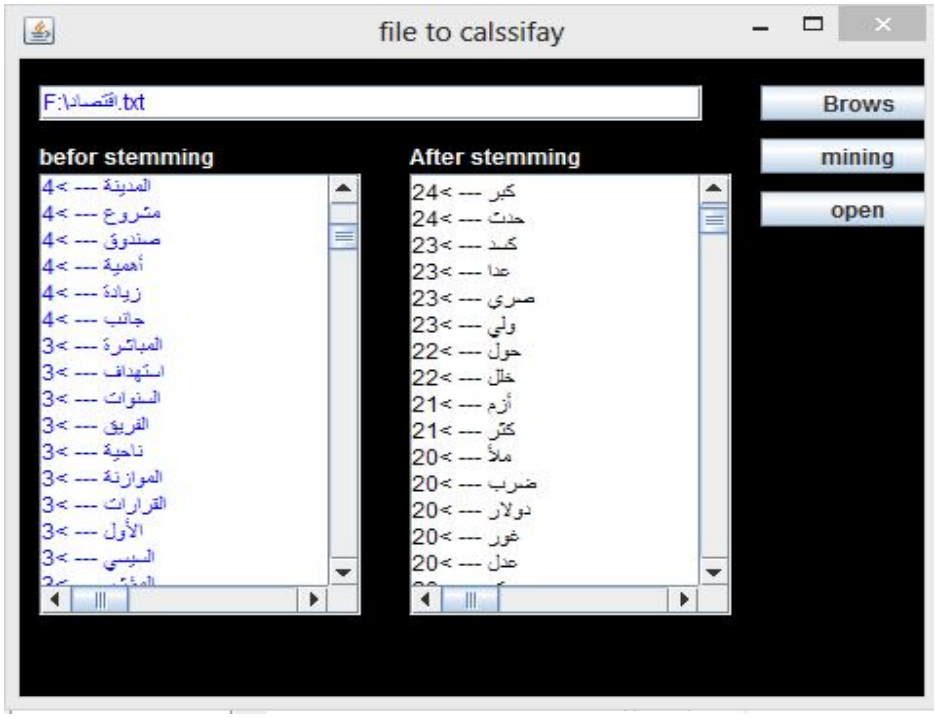


Figure ٥.٦(٣)

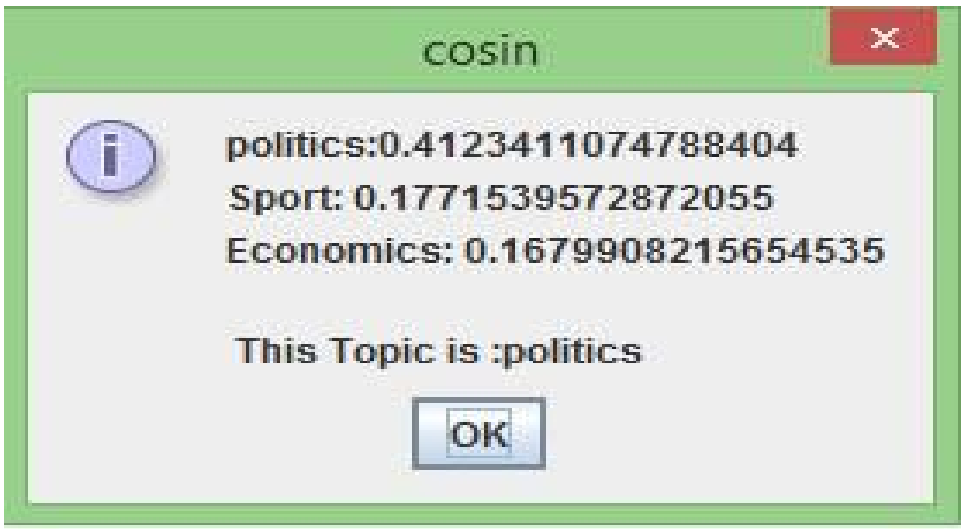


Figure ٥.٦ (٤)

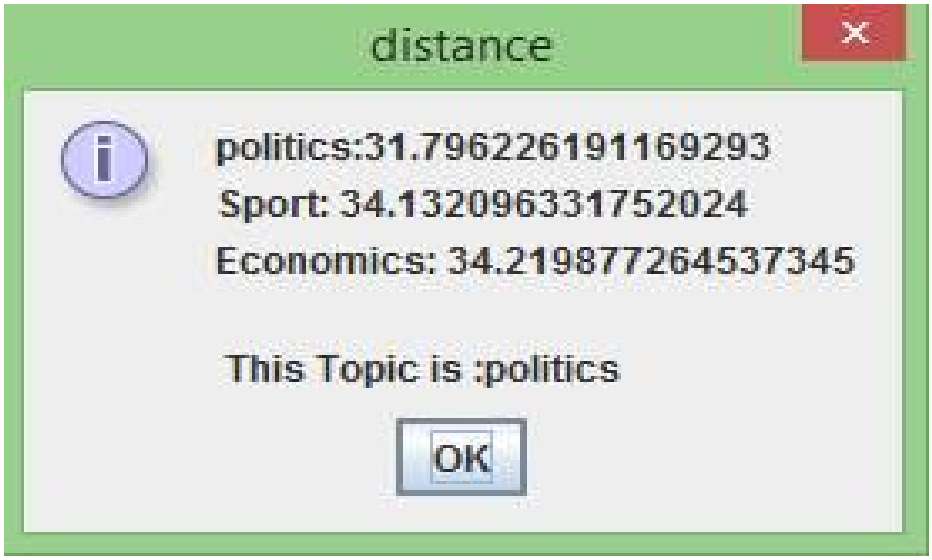


Figure ٥.٦ (٥)

٥.٧ Twitter Data Collector Tool

This tool extract data from Twitter.

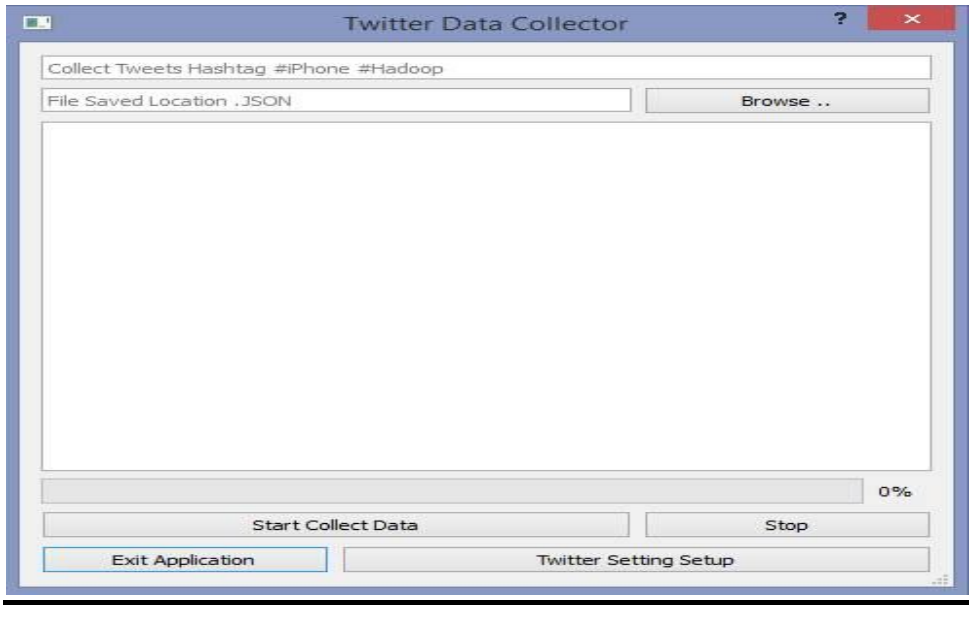


Figure ٥.٧(١) Twitter data collector tool

But its output is encrypted.

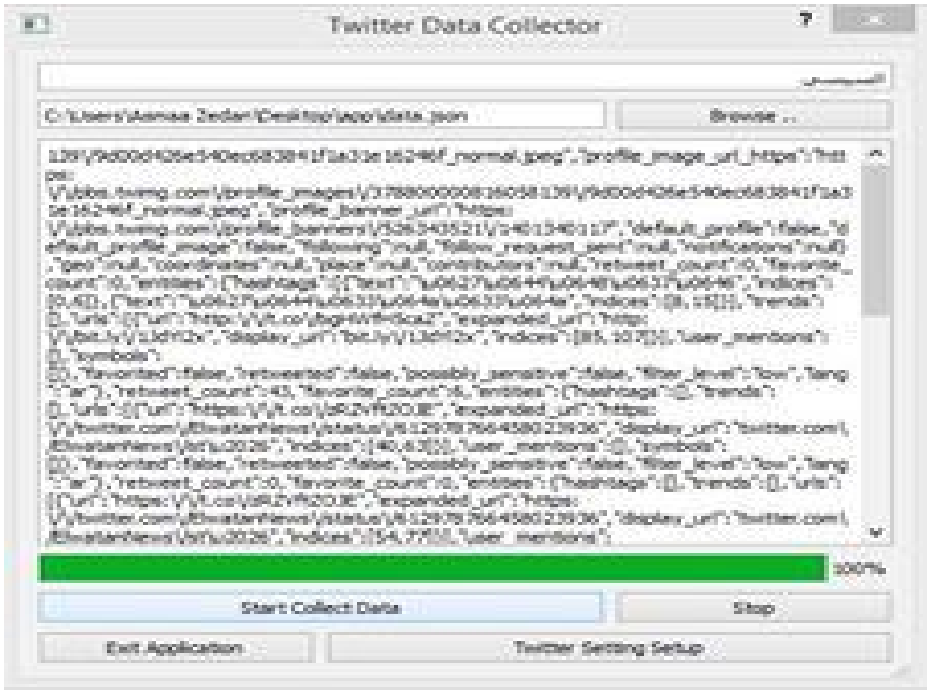


Figure ٥.٧(٢) decrypted result

<u>To solve this problem we use java code to decrypt.</u>

```
package projectencoding;
import java.io.*;
import java.nio.file.Files;
import java.nio.file.Paths;
import java.util.regex.Matcher;
import java.util.regex.Pattern;

publicclass Projectencoding {


publicstaticvoid main(String[]args)throws UnsupportedEncodingException,
IOException {
try
        {
```

```
String content = new String(Files.readAllBytes(Paths.get("C:\\Users\\AsmaaZedan\\Desktop\\app\\data.json")),"UTF-٨");

BufferedWriter out = new BufferedWriter (new OutputStreamWriter (
 new FileOutputStream ("C:\\Users\\Asmaa Zedan\\Desktop\\out.txt"), "UTF-٨"));
 BufferedWriter out٢ = new BufferedWriter (new OutputStreamWriter(
        new FileOutputStream ("C:\\Users\\Asmaa Zedan\\Desktop\\out٢.txt"), "UTF-٨"));

  // Pattern p = Pattern.compile ("\"text\":\"(.*?)\",\"source\"");
  Pattern p = Pattern.compile ("\"text\":\"(.*?)\",\"source\"");
  Pattern p٣ = Pattern.compile ("\\{(.*?)\\}");
  Pattern p٢ = Pattern.compile ("^\"text\":\"(.*?)\",\"source\"");
 Pattern p٤ = Pattern.compile ("\"created_at\":\"(.*?)\",");

Matcher m = p.matcher (content);

Matcher m٢ = p٢.matcher (content);
Matcher m٣ = p٣.matcher (content);
Matcher m٤ = p٤.matcher (content);
String str٣ = null;

String str٢;
while(m٤.find())
          {
while (m٣.find()) {

while(m.find()) {

            String str=m.group(١);
int indexname= str.indexOf("@");
```

```
        int index= str.indexOf("\\")          ١٤٠
           str٢=str.substring(index, str.length());
           String str٤=m٤.group(١);
           out.write(str٤);
           out.newLine();
if(indexname!=-١){
            str٣=str.substring(indexname,index);
           out.write(str٣);

           out.newLine();
        }
           str٢ = str٢.replace("\\","");
            String[] arr = str٢.split("u");

              String text = "";
        for(int i = ١; i < arr.length; i++){
                String temp=arr[i];
        if(arr[i].length()>٤)
                {
                    temp=arr[i].substring(٠, ٤);
                }
        int hexVal = Integer.parseInt(temp, ١٦);
              text += (char)hexVal;
              text+=arr[i].substring(٤,arr[i].length());
}


        out.write(text);

        out.newLine();
         out٢.write(text);
         out٢.newLine();
         out.flush();
```

```
            out٢.flush();                                    ١٤١
        }
}
   }}
catch(IOException | NumberFormatException ex)
        {
            System.out.println (ex.getMessage());
        }
    }
}
```

Figure ٥.٧ result after decryption

٥.٨ Conclusion

In this chapter we discuss data mining and its role in this project, mining seems to be the backbone of our project as it is the first step without it, we cannot began this project.

Chapter 6

ic Web

٦.١ Introduction

٦.٢ The Development Methodology

٦.٣ Installation Tool Techniques

٦.٤ SPARQL Query

٦.١ Introduction

The **Semantic Web** is an extension of the web through standards by world wed web (W٣C).

According to the W٣C, "The Semantic Web provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries.

When applied in the context of the modern internet, it extends the network of hyperlinked human-readable web pages by inserting machine-readable metadata about pages and how they are related to each other. This enables automated agents to access the Web more intelligently and perform more tasks on behalf of users.

Semantic Web involves publishing in languages specifically designed for data: Resource Description Framework (RDF), Web Ontology Language (OWL), and Extensible Markup Language (XML). HTML describes documents and the links between them. RDF, OWL, and XML, by contrast, can describe arbitrary things such as people, meetings, or airplane parts.

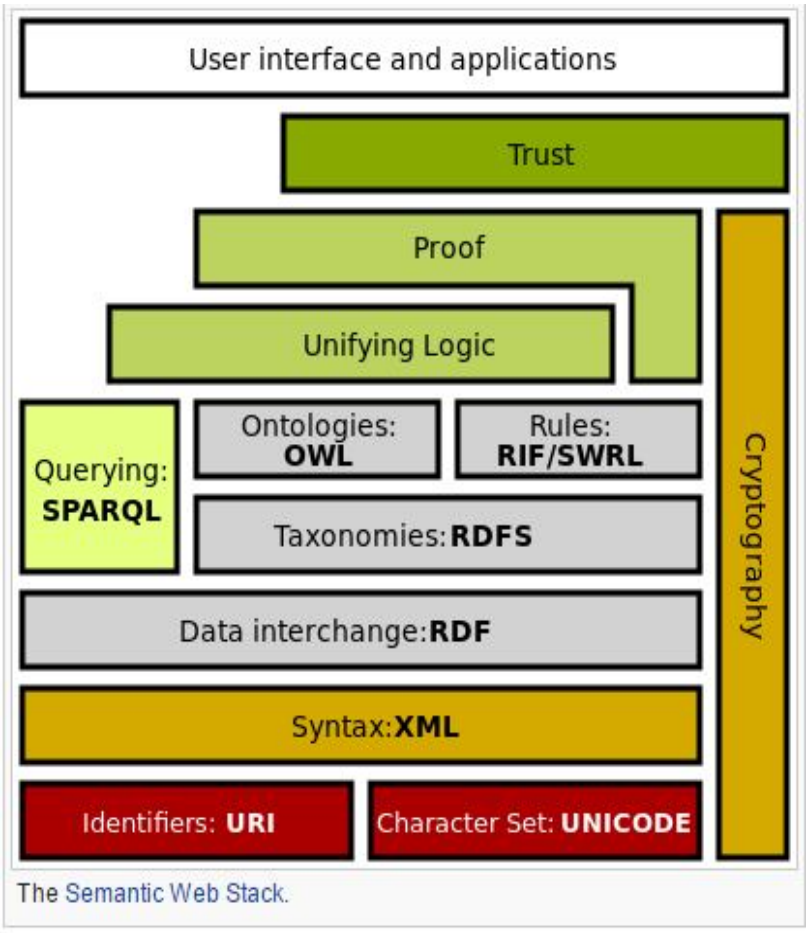


Figure ٦.١ Semantic Web Stack

Our system is based on ontology.

٦.٢ The Development Methodology

Ontology is the philosophical study of the nature of being, becoming, existence, or reality, as well as the basic categories of being and their relations.

Traditionally listed as a part of the major branch of philosophy known as metaphysics, ontology deals with questions concerning what entities exist or may be said to exist, and how such entities may be grouped, related within a hierarchy, and subdivided according to similarities and differences.

Ontology is the backbone of semantic web.

The main thread of ontology in the philosophical sense is the study of entities and their relations. The question ontology asks is: What kinds of things exist or can exist in the world, and what manner of relations can those things have to each other? Ontology is less concerned with what is than with what is possible.

Ontologies are also considered one of the pillars of the Semantic Web, although they do not have a universally accepted definition. A (Semantic Web) vocabulary can be considered as a special form of (usually light-weight) ontology, or sometimes also merely as a collection of URIs with an (usually informally) described meaning.

Which ontology editor we will use?

We will use **protégé** as a tool to create our system ontology.

Protégé is a free, open source ontology editor and a knowledge acquisition system. Protégé provides a graphic user interface to define Ontologies. It also includes deductive classifiers to validate that models are consistent and to infer new information based on the analysis of ontology. Like Eclipse, Protégé is a framework for which various other projects suggest plugins. This application is written in Java and heavily uses Swing to create the rather complex user interface. Protégé recently has over ٢٠٠,٠٠٠ registered users.

Ontologies range from taxonomies, classification, database schemas to fully axiomatized theories. Protégé Desktop supports creation and editing of one or more ontologies in a single workspace via a completely customizable user interface.

Visualization tools allow for interactive navigation of ontology relationships. Advanced explanation support aids in tracking down inconsistencies. Refactor operations available including ontology merging, moving axioms between ontologies, rename of multiple entities, and more.

٦.٣ Installation Tool Techniques

٦.٣.١ Install Protégé

Protégé installation: We will setup and work on ٣.٤.٨ version of protégé Go to: http://protege.stanford.edu/ to download protégé then start to install it.

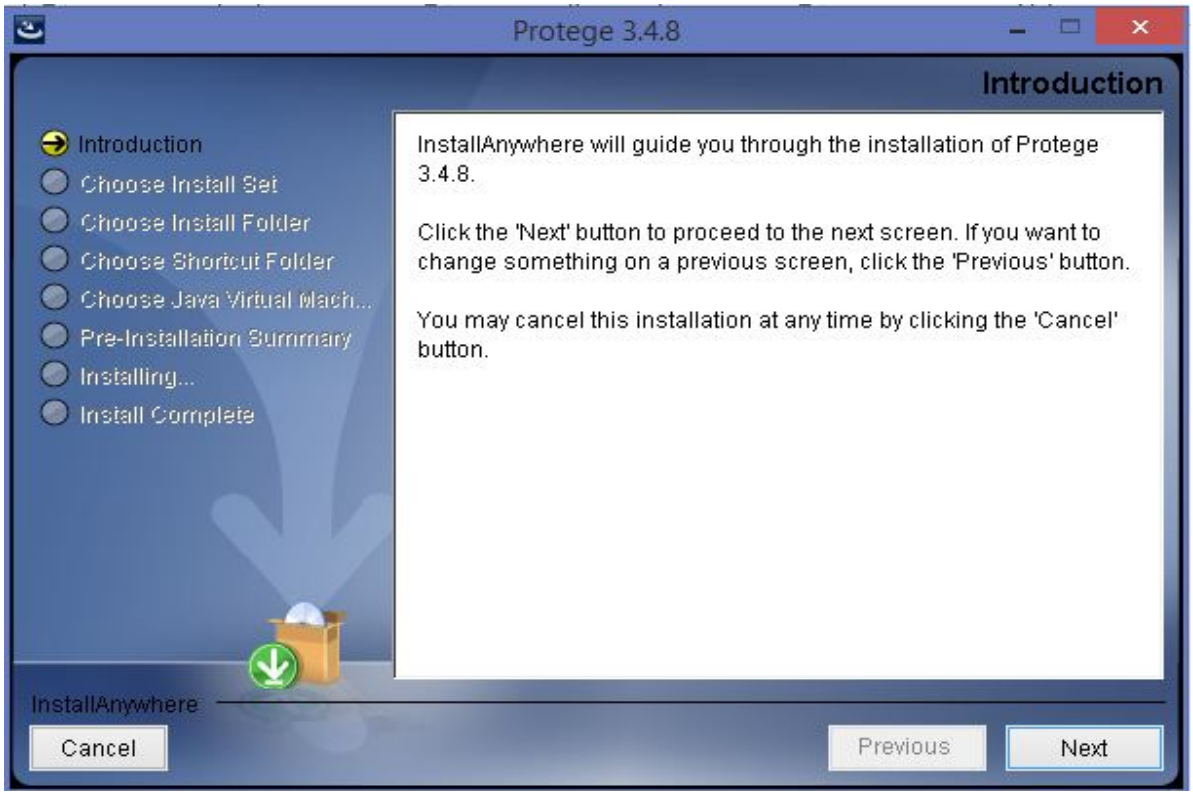


Figure ٦.٣.١ Install Protégé (١)

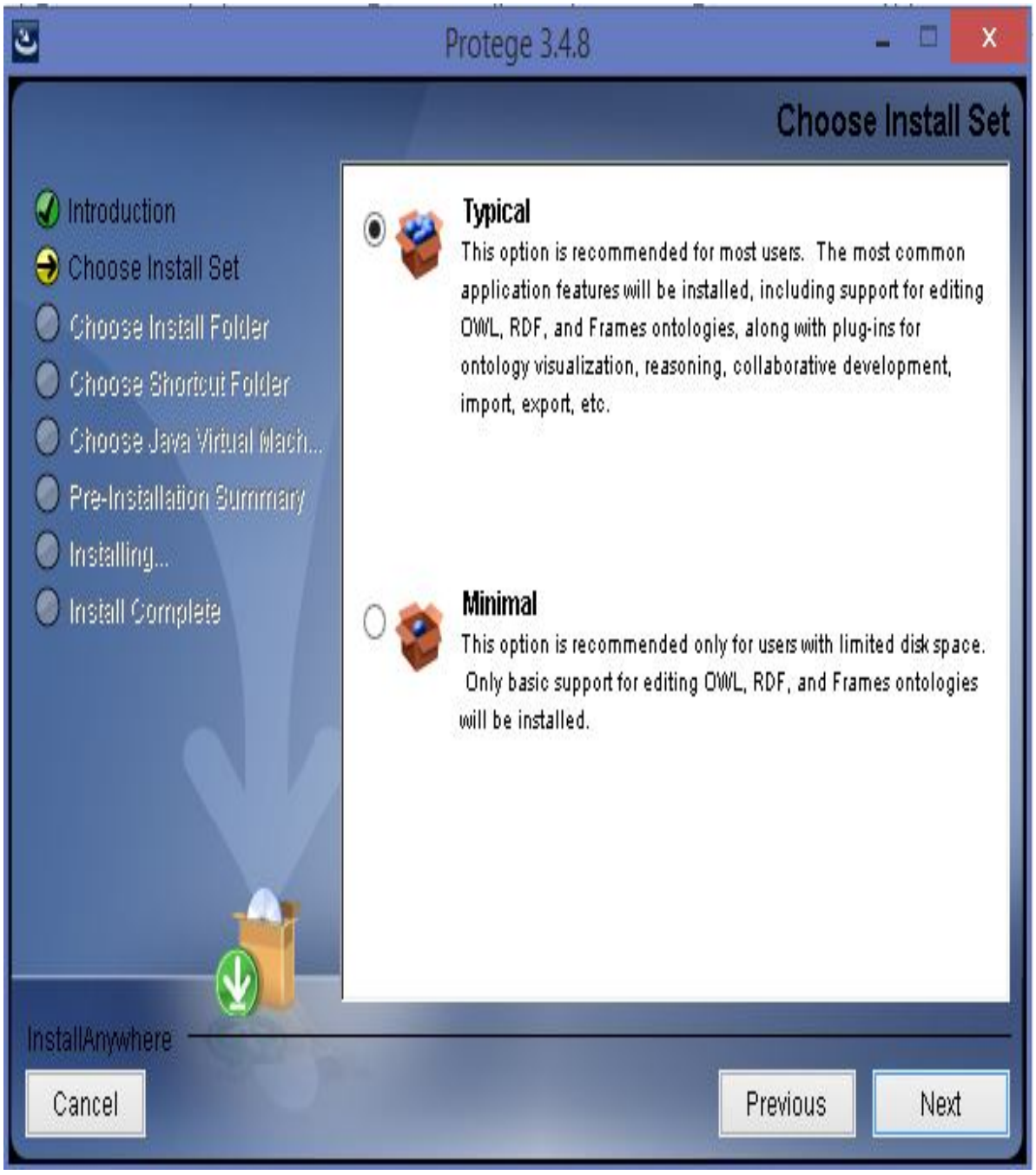


Figure ٦.٣.١ Install Protégé (٢)

Figure ٦.٣.١ Install Protégé (٣)

Figure ٦.٣.١ Install Protégé (٤)

Figure ٦.٣.١ Install Protégé (٥)

Figure ٦.٣.١ Install Protégé (٦)

٦.٣.٢ Creating Ontology

After installation run the program and start creating your ontology

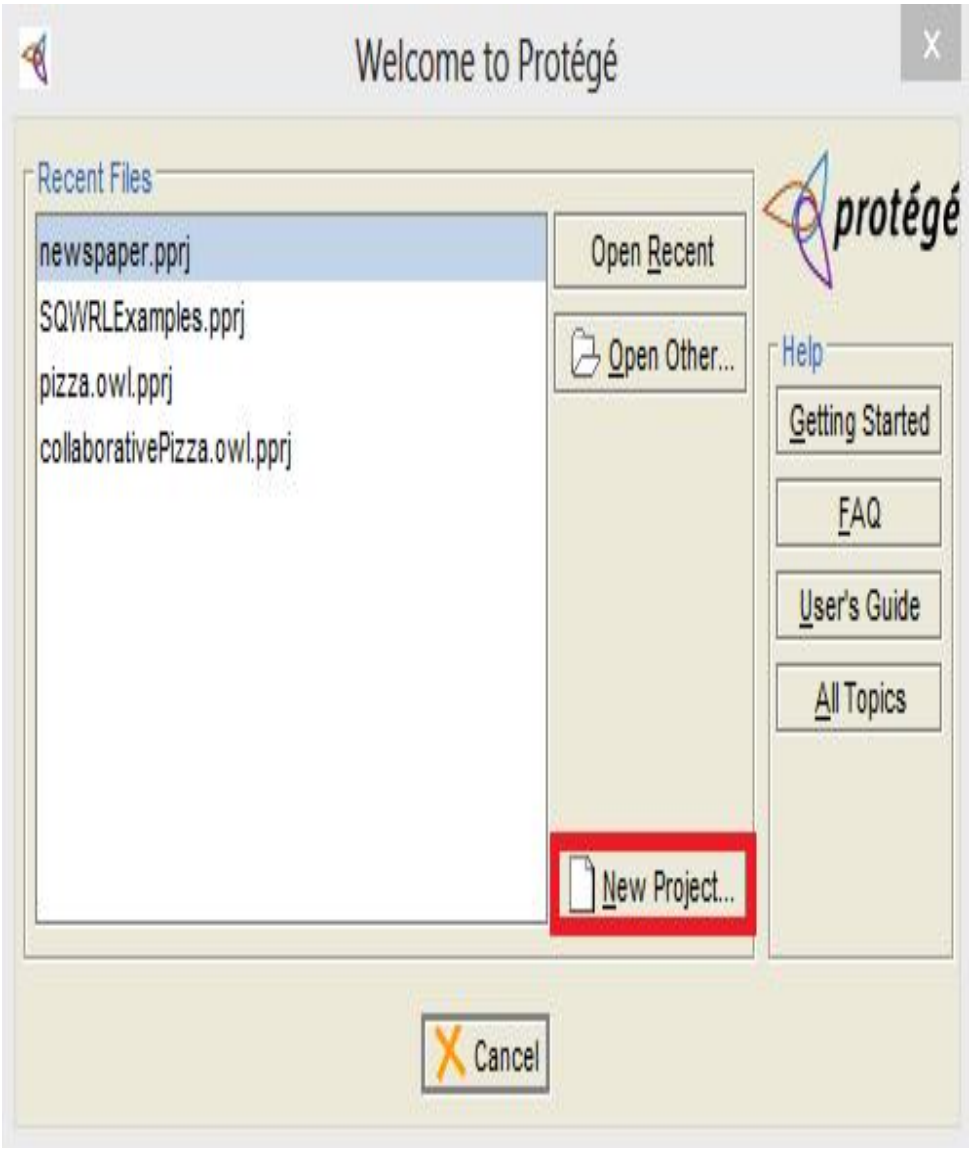


Figure ٦.٣.٢ Create Ontology (١)

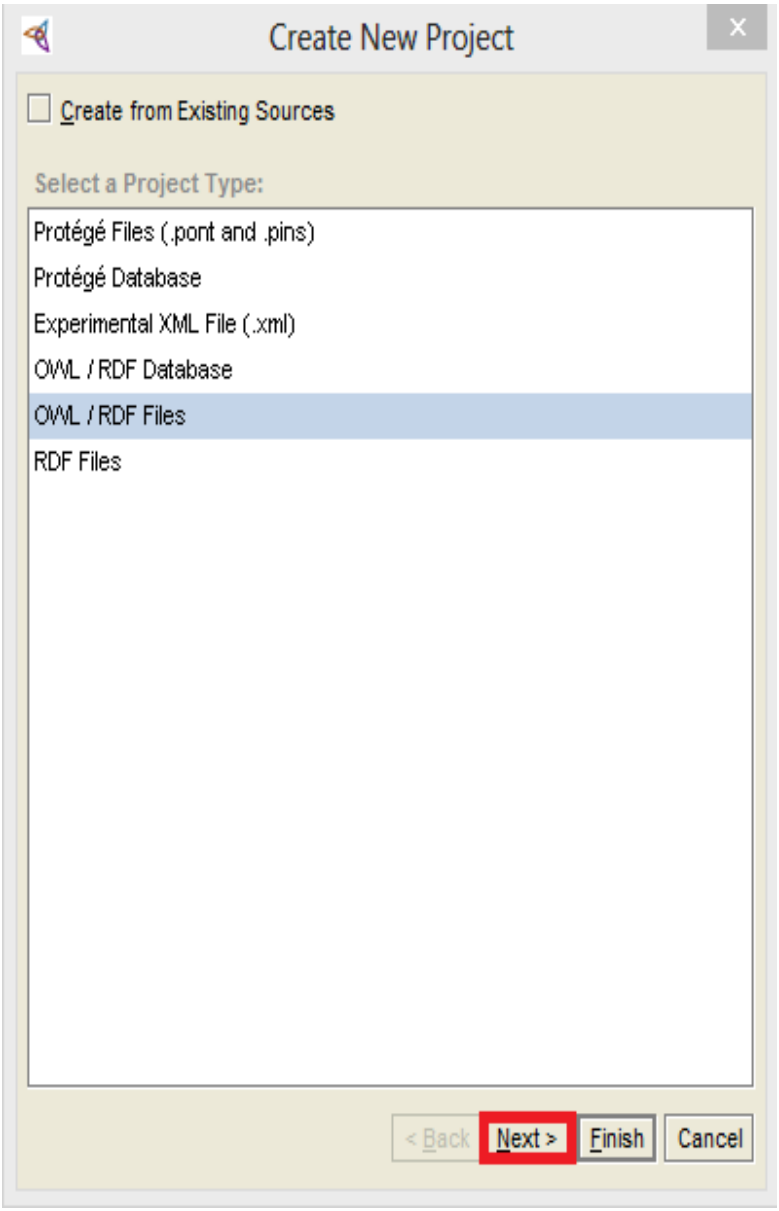


Figure ٦.٣.٢ Create Ontology (٢)

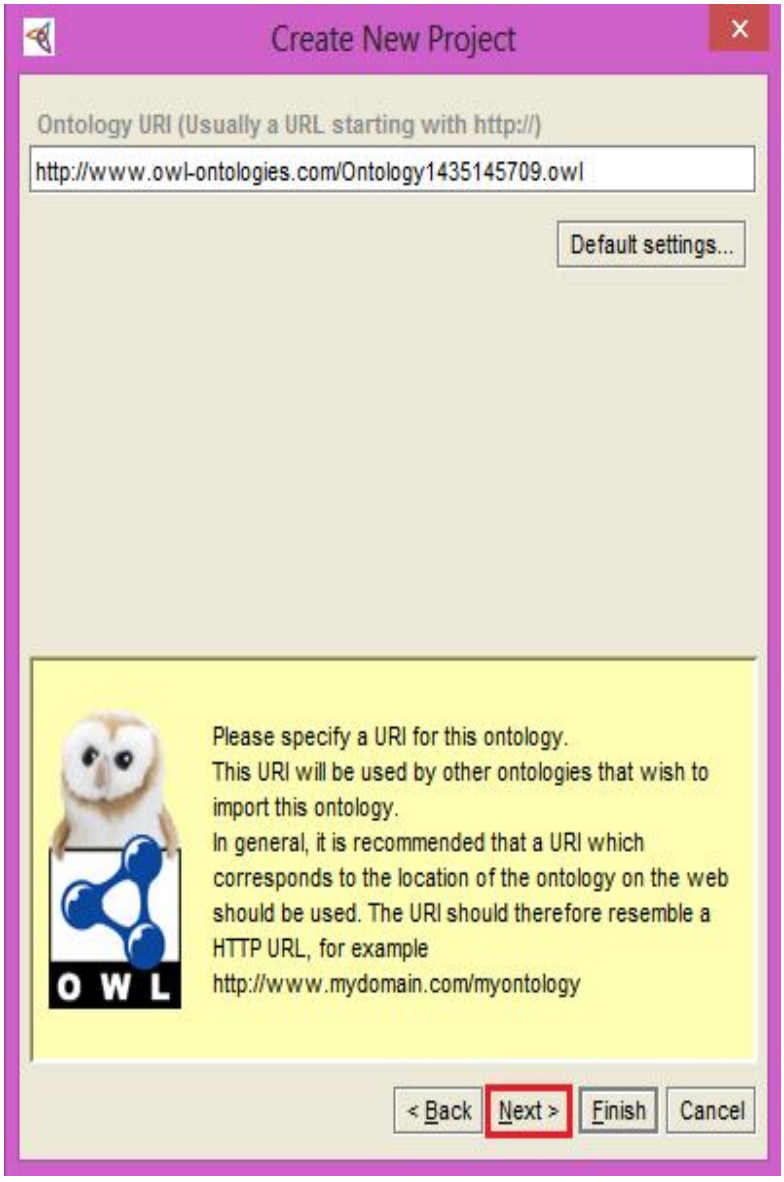


Figure ٦.٣.٢ Create Ontology (٣)

Figure ٦.٣.٢ Create Ontology (٤)

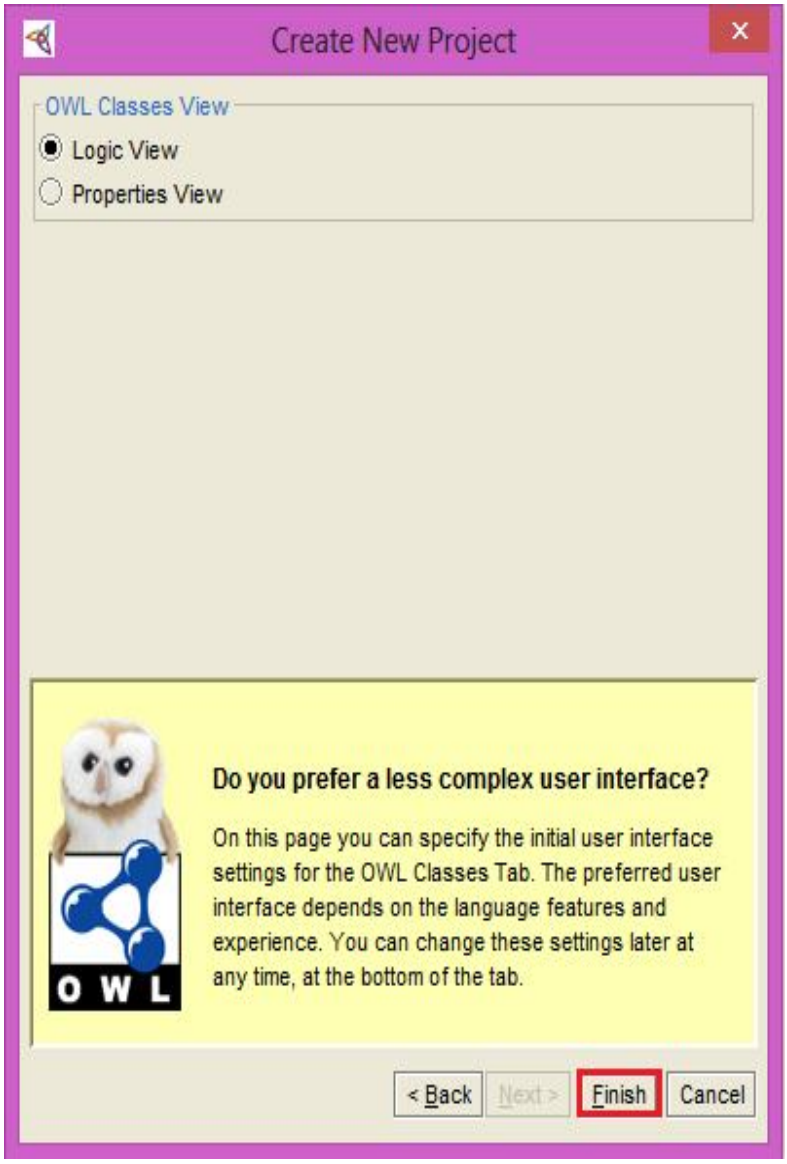


Figure ٦.٣.٢ Create Ontology (٥)

Then start to create our classes, properties, instances and rules.

Our ontology classes are divided into three classes' positive, negative and balanced categories of people according to government opinion.

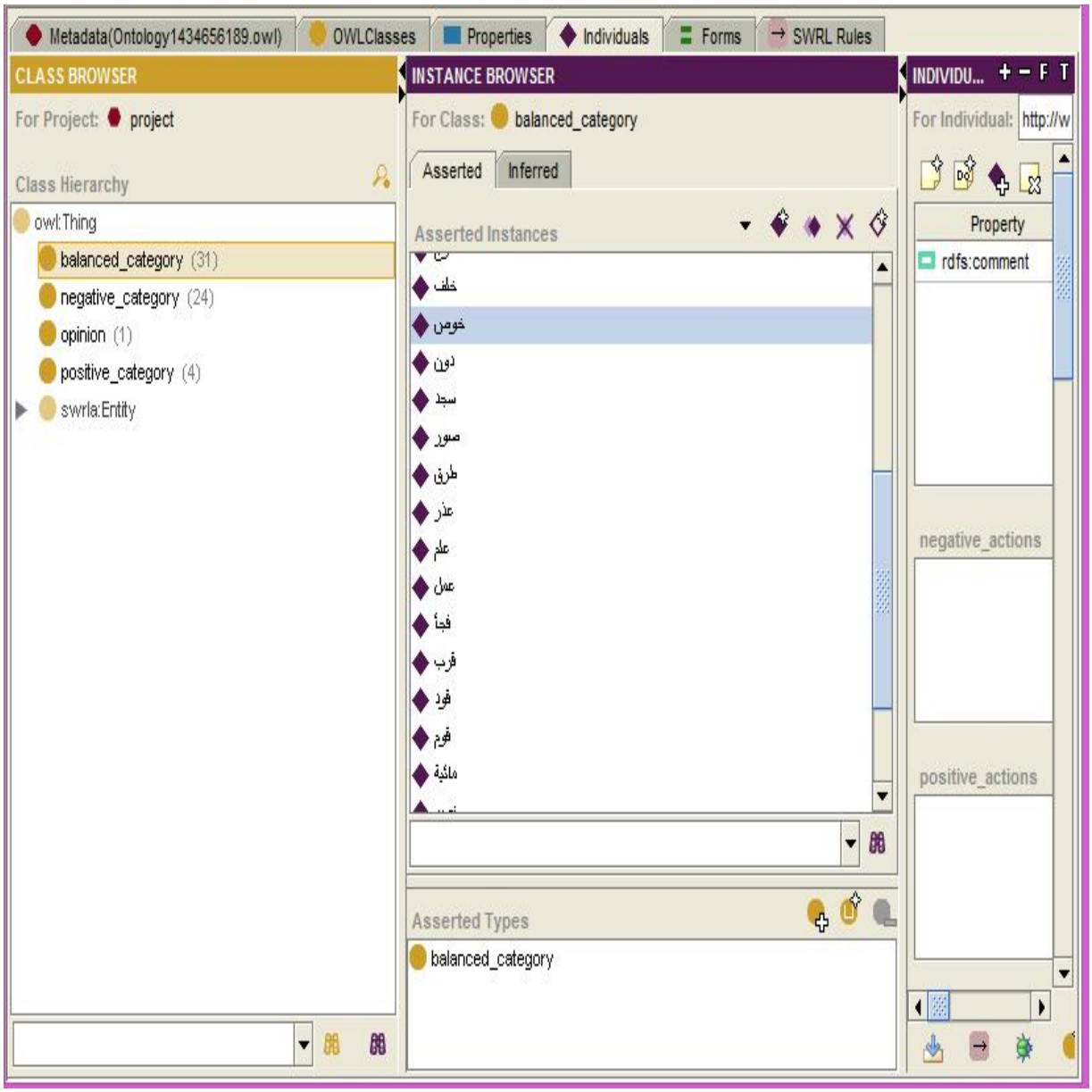


Figure ٦.٣.٢ Create Ontology (٦)

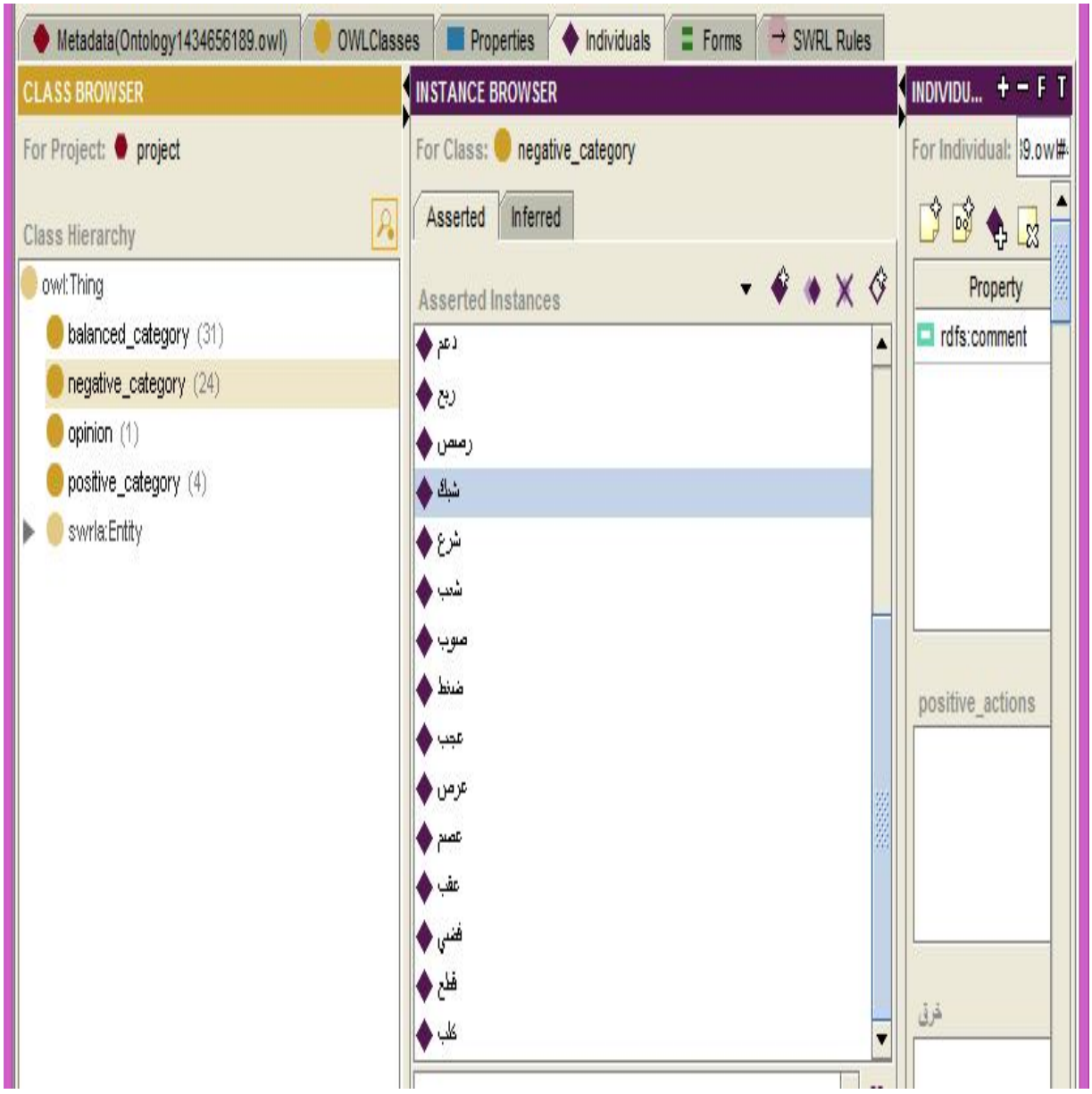


Figure ٦.٣.٢ Create Ontology (٧)

In object properties we divided actions into positive, negative and balanced actions.

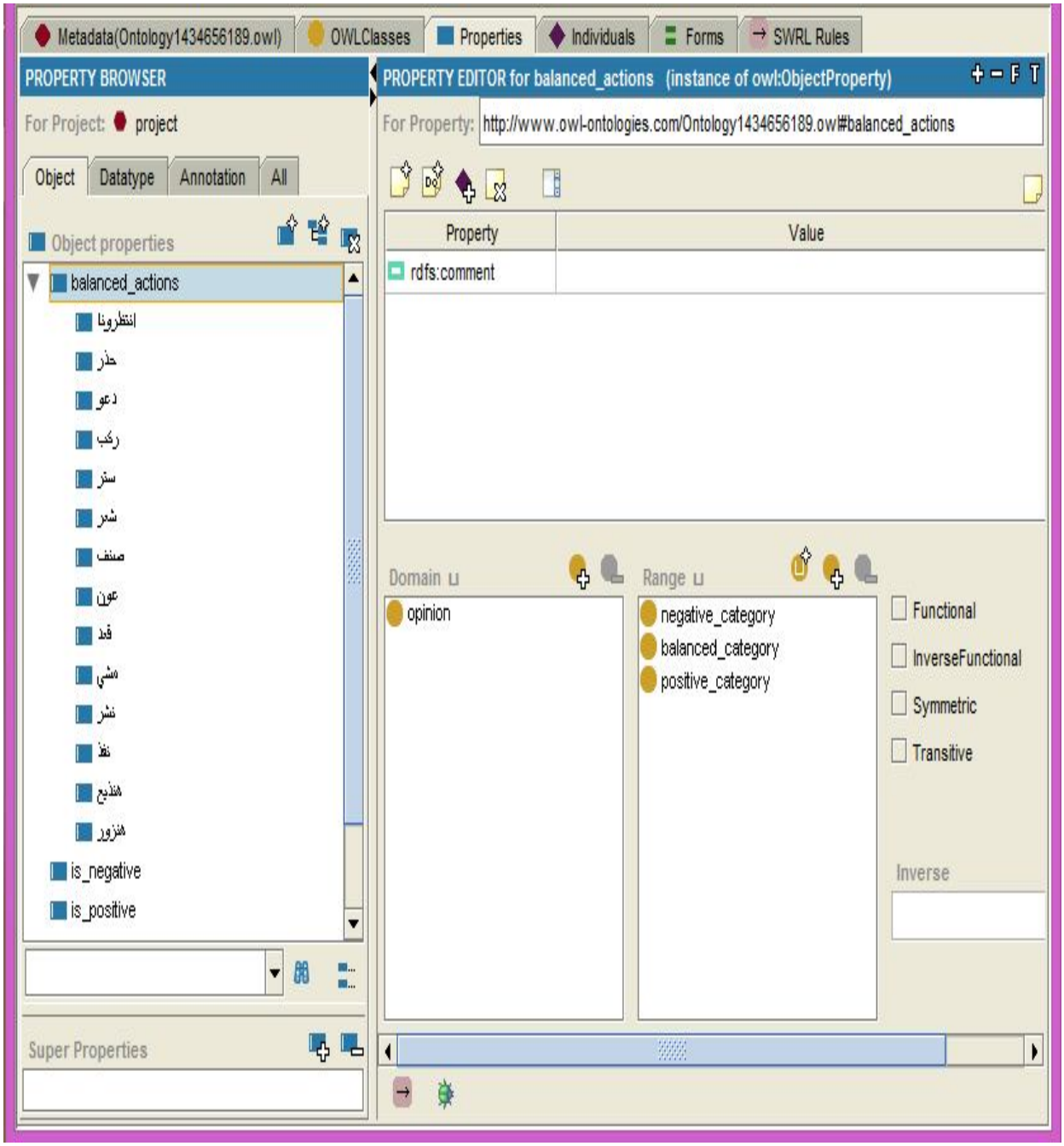


Figure ٦.٣.٢ Create Ontology (٨)

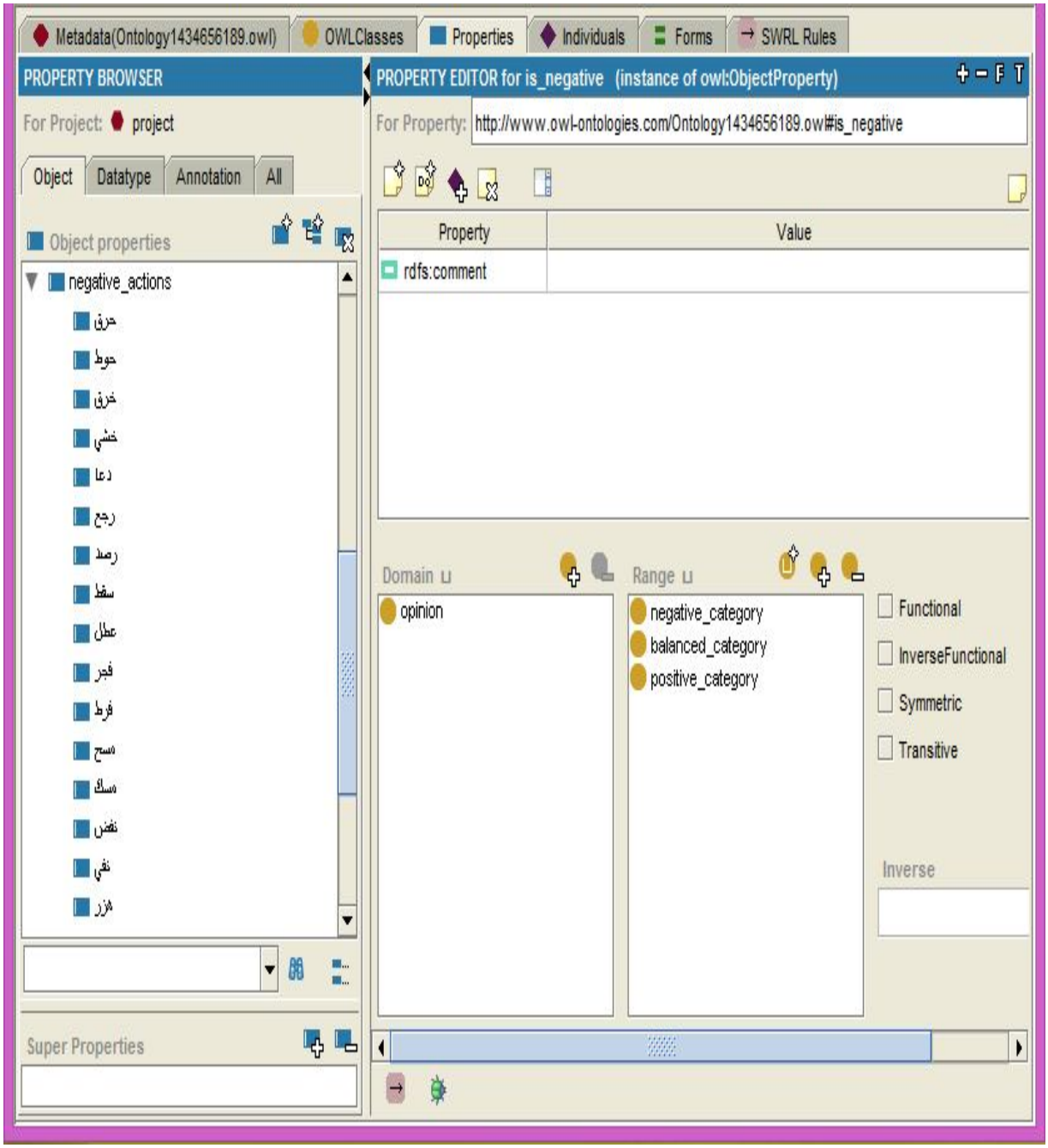


Figure ٦.٣.٢ Create Ontology (٩)

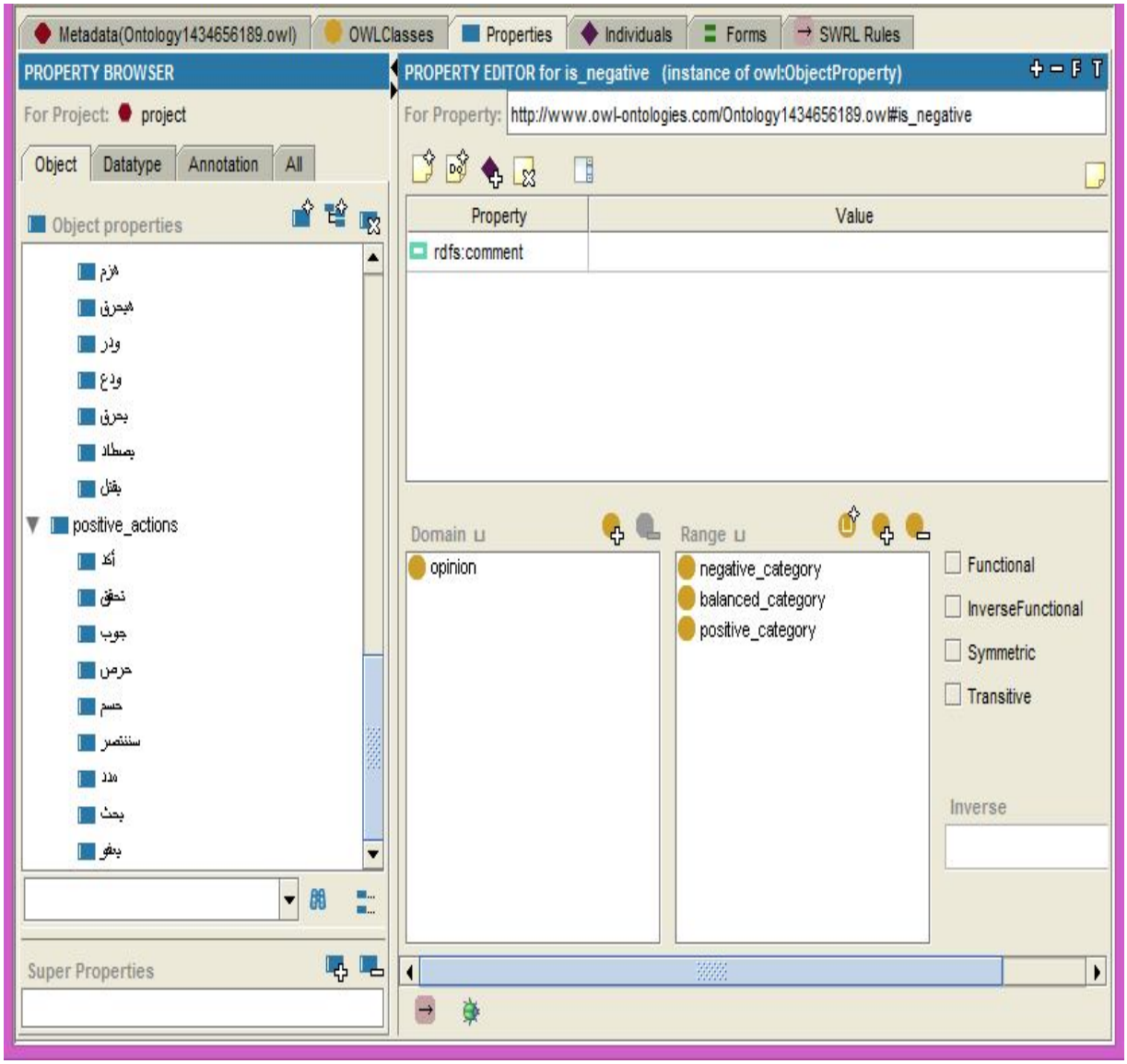


Figure ٦.٣.٢ Create Ontology (١٠)

Note: all words classes, properties and instances put into ontology after making stemming for it.

The RDF graph model:

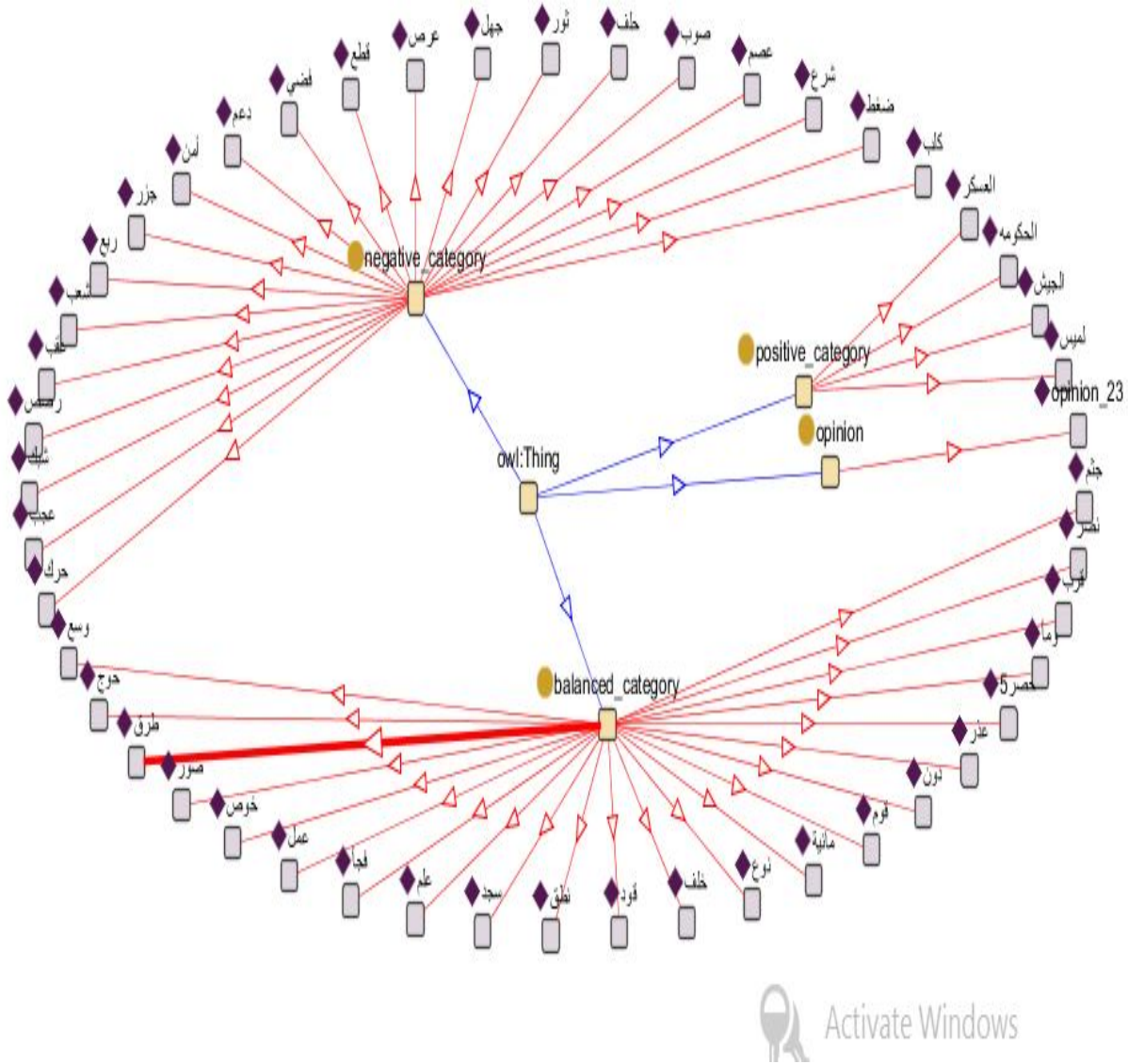


Figure ٦.٣.٢ RDF Graph

٦.٣.٣ Install Jess and Building SWRL

Semantic Web Rule Language

The Semantic Web Rule Language (SWRL) is a proposed language for the Semantic Web that can be used to express rules as well as logic, combining OWL DL or OWL Lite with a subset of the Rule Markup Language (itself a subset of Data log).

The specification was submitted in May ٢٠٠٤ to the W٣C by the National Research Council of Canada, Network Inference (since acquired by web Methods), and Stanford University in association with the Joint US/EU ad hoc Agent Markup Language Committee. The specification was based on an earlier proposal for an OWL rules language.

SWRL has the full power of OWL DL, but at the price of decidability and practical implementations. However, decidability can be regained by restricting the form of admissible rules, typically by imposing a suitable safety condition.

Rules are of the form of an implication between an antecedent (body) and consequent (head). The intended meaning can be read as: whenever the conditions specified in the antecedent hold, then the conditions specified in the consequent must also hold.

١. Open jess and SWRL taps as:

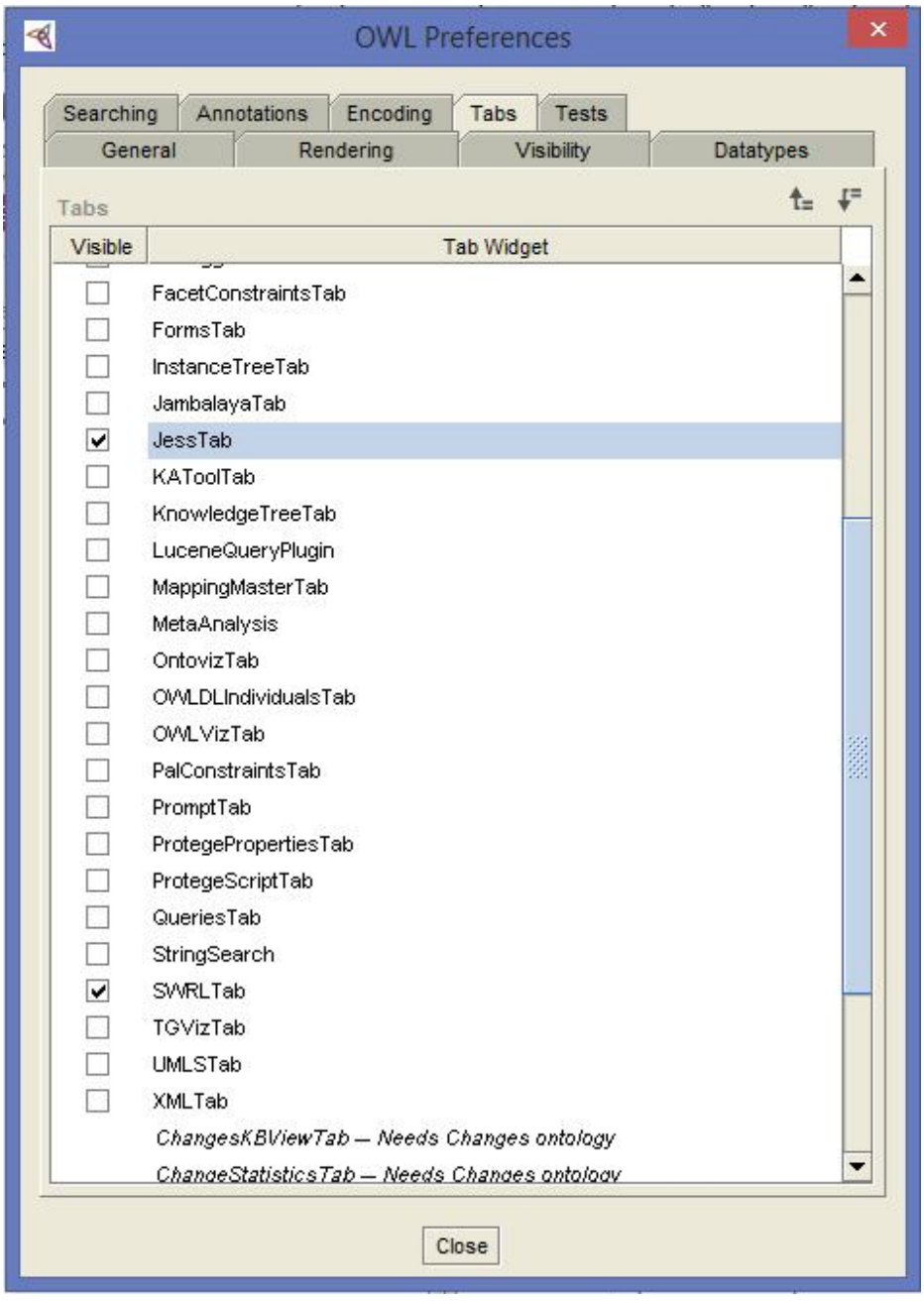
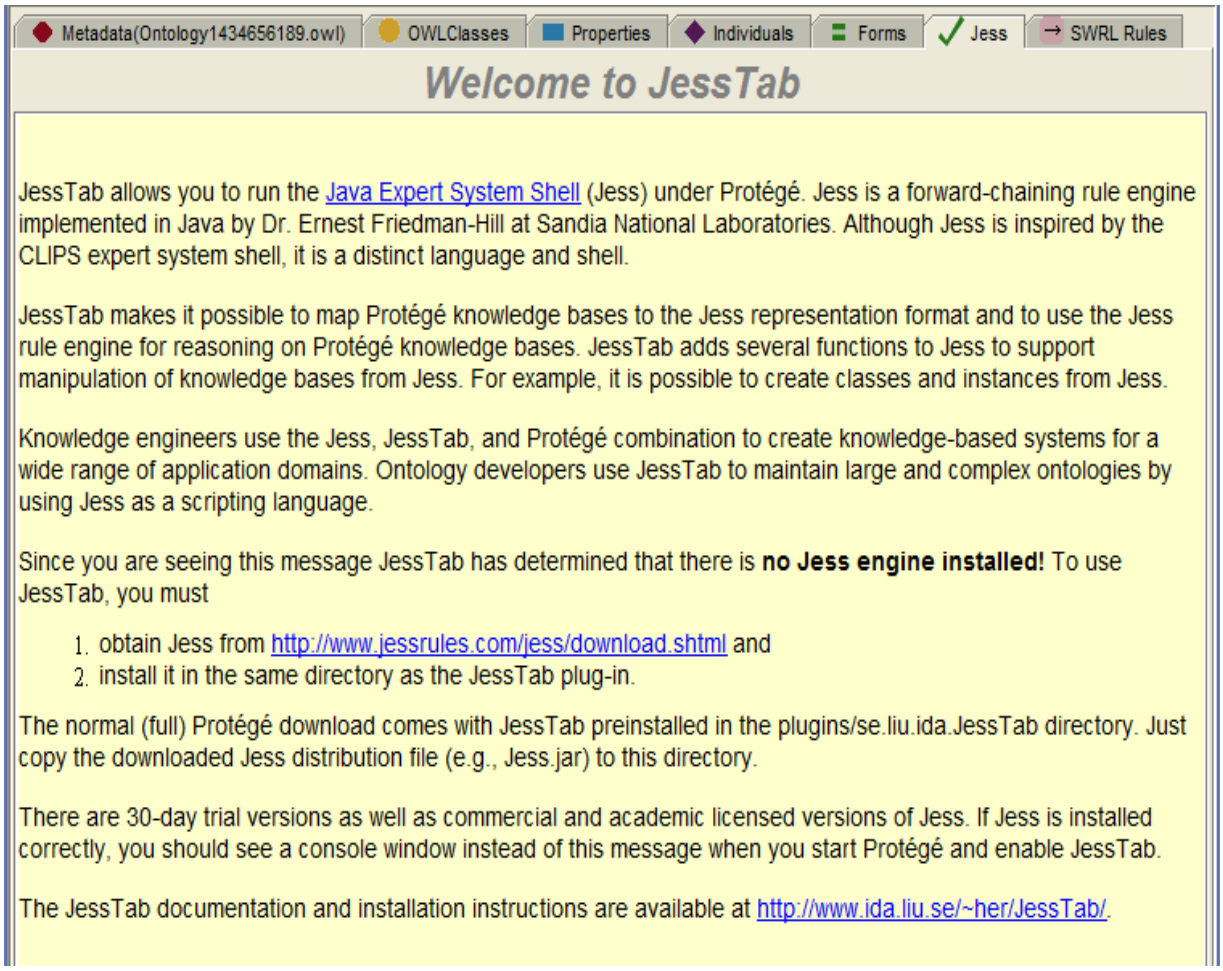
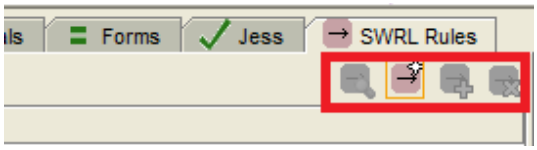


Figure ٦.٣.٣ Jess Preferences

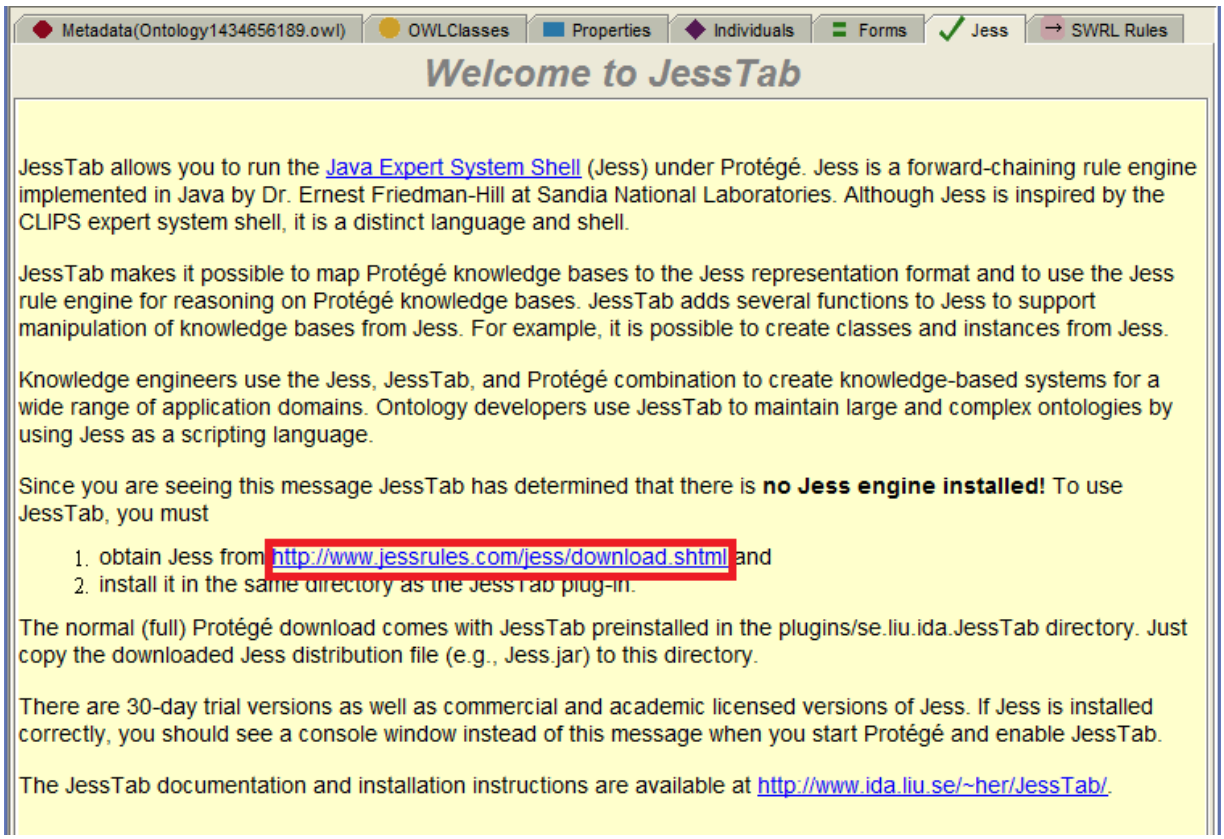
Jess Tab:



SWRL tab:



٢. Now we want to download and add jess engine:
download it from



٣. Then extract files from 

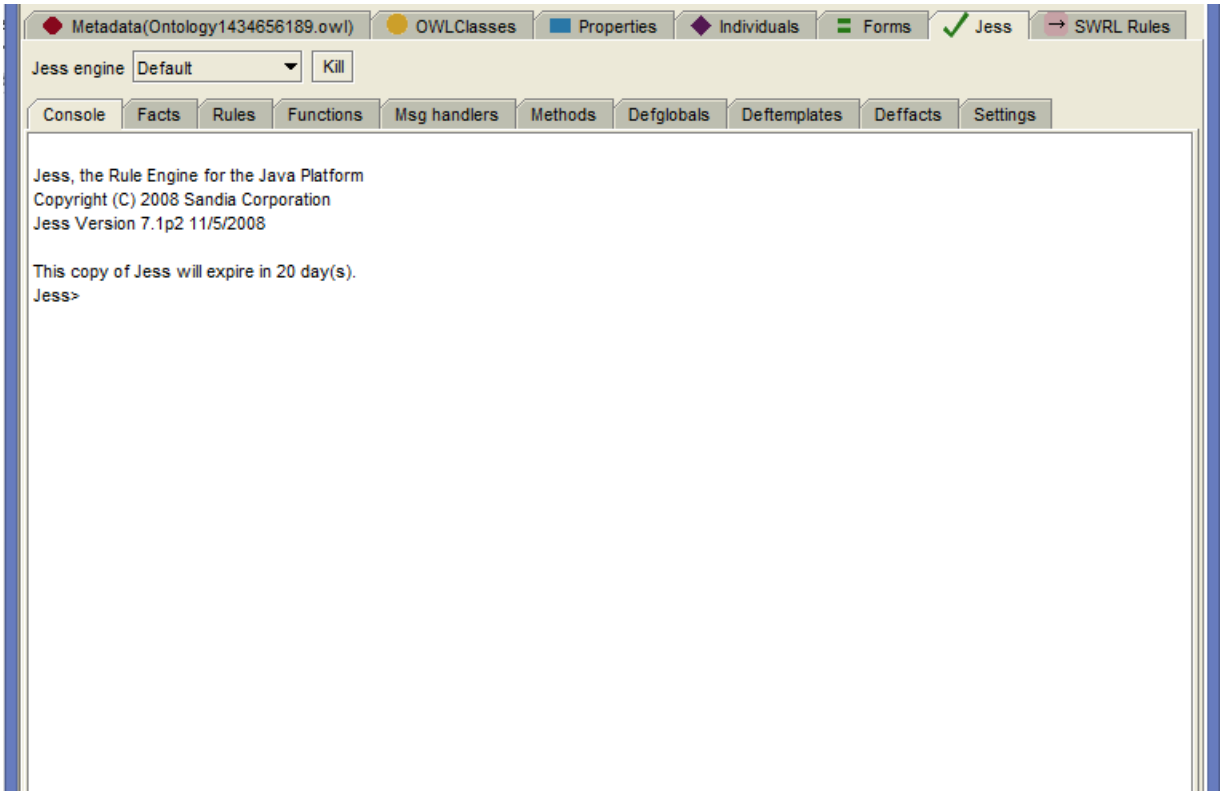
Open: Jess٧١p٢\lib

٤. Then take jess.jar copy and paste it into:

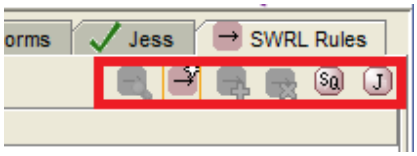
C: program file \Protege_٣.٤.٨ \plugins \edu.stanford.smi.protegex.owl

٥. Restart protégé

٦. If it work correctly jess tap will appear as



And SWRL will appear as:



٧. Then start to add your rules:

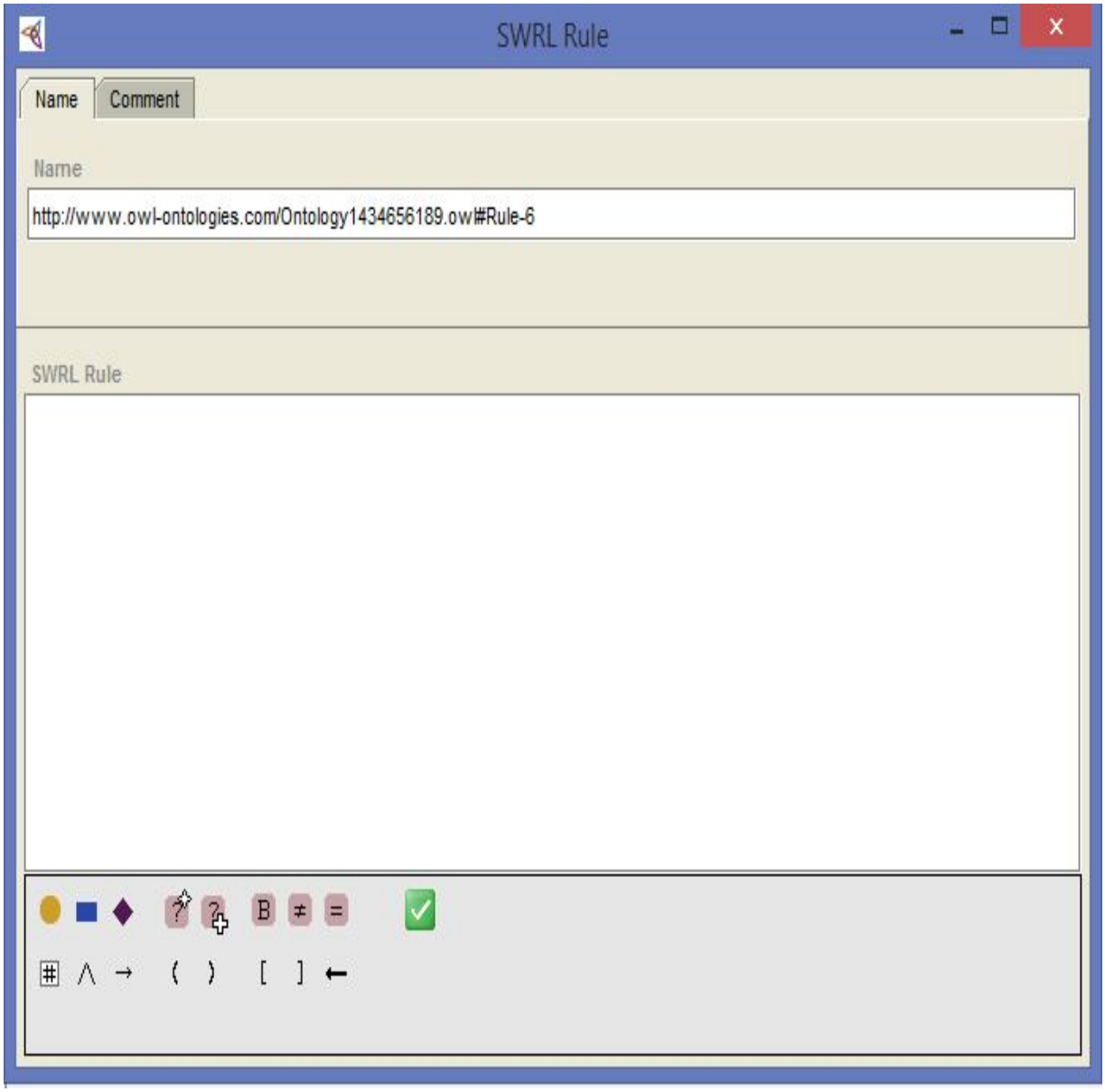


Figure ٦.٣.٣ SWRL Rule

Rule	Result
positive actions(?f, ?l) ∧positive category(?f) ∧ balanced category(?l) الجيش يدعم المواطنين	Is positive
positive actions(?f, ?l) ∧ positive category(?f) ∧positive category(?l) الجيش يدعم الاستقرار	Is positive
positive actions(?f, ?l) ∧ positive category(?f) ∧negative category(?l) الجيش يدعم الفساد	Is positive
negative actions(?f, ?l) ∧ positive category(?f) ∧ balanced category (?l) الجيش المصرى يقتل المواطنين	In negative
negative actions(?f, ?l) ∧ positive category(?f) ∧ positive category (?l) الجيش المصرى يقتل مجنديه	Is negative
negative actions(?f, ?l) ∧ positive category(?f) ∧ negative category (?l) الجيش المصرى يقتل الارهابيين	Is positive
balanced actions(?f, ?l) ∧ positive category(?f) ∧ balanced category(?l)	Is balanced
balanced actions(?f, ?l) ∧ positive category(?f) ∧positive category(?l) الجيش المصرى يدعو للصلح	Is positive
balanced actions(?f, ?l) ∧ positive category(?f) ∧negative category(?l) الجيش المصرى يدعو للانتقام من الاخوان	Is positive
positive actions(?f, ?l) ∧negative category(?f) ∧ balanced category(?l) الاخوان يدعمون المواطنين	Is positive
positive actions(?f, ?l) ∧negative category(?f) ∧positive category(?l) الاخوان يدعمون النهضه	Is positive
positive actions(?f, ?l) ∧negative category(?f) ∧negative category(?l) الاخوان يدعمون الارهاب	Is negative
negative actions(?f, ?l) ∧negative category(?f) ∧ balanced category (?l) الاخوان يقتلون المواطنين	Is negative
negative actions(?f, ?l) ∧negative category(?f) ∧ positive category (?l) الاخوان يقتلون الجنود	Is negative
negative actions(?f, ?l) ∧negative category(?f) ∧ negative category (?l) الاخوان يقتلون الفاسدين	Is negative
balanced actions(?f, ?l) ∧ negative category(?f) ∧ balanced category(?l)	Is balanced
balanced actions(?f, ?l) ∧negative category(?f) ∧positive category(?l) الاخوان يدعون للصلح	Is positive

balanced actions(?f, ?l) ∧negative category(?f) ∧negative category(?l) الاخوان يدعون للانتقام	Is negative
positive actions(?f, ?l) ∧balanced category(?f) ∧ balanced category(?l) الشعب يدعم فريقه القومى	Is positive
positive actions(?f, ?l) ∧balanced category(?f) ∧positive category(?l) الشعب يدعم الجيش	Is positive
positive actions(?f, ?l) ∧balanced category(?f) ∧negative category(?l) الشعب يدعم الاخوان	Is negative
negative actions(?f, ?l) ∧balanced category(?f) ∧ balanced category (?l) الشعب يخرب المنشئات	Is negative
negative actions(?f, ?l) ∧balanced category(?f) ∧ positive category (?l) الشعب يقتل الجنود	Is negative
negative actions(?f, ?l) ∧balanced category(?f) ∧ negative category (?l) الشعب يقاتل الاعداء	Is positive
balanced actions(?f, ?l) ∧balanced category(?f) ∧ balanced category(?l)	Is balanced
balanced actions(?f, ?l) ∧balanced category(?f) ∧positive category(?l) الشعب يدعو للاصلاح	Is positive
balanced actions(?f, ?l) ∧balanced category(?f) ∧negative category(?l) الشعب يدعو للفساد	Is balanced

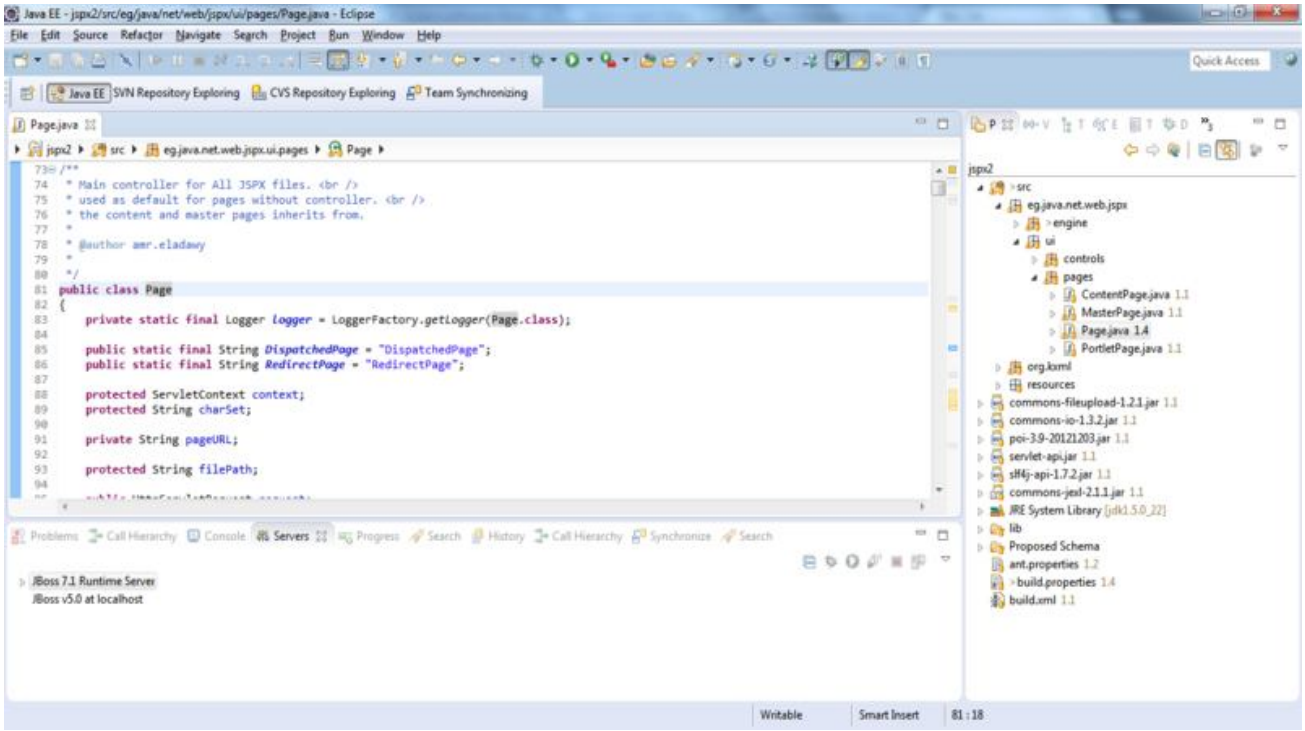
٦.٣.٤ Install and Run Jena

Eclipse (software)

In computer programming, Eclipse is an integrated development environment (IDE). It contains a base workspace and an extensible plug-in system for customizing the environment. Written mostly in Java, Eclipse can be used to develop applications.

By means of various plug-ins, Eclipse may also be used to develop applications in other programming languages: Ada, ABAP, C, C++, COBOL, Fortran, Haskell, JavaScript, Lasso, Lua, Natural, Perl, PHP, Prolog, Python, R, Ruby (including Ruby on Rails framework), Scala, Clojure, Groovy, Scheme, and Erlang. It can also be used to develop packages for the software Mathematica. Development environments include the Eclipse Java development tools (JDT) for Java and Scala, Eclipse CDT for C/C++ and Eclipse PDT for PHP, among others.

The initial codebase originated from IBM Visual Age. The Eclipse software development kit (SDK), which includes the Java development tools, is meant for Java developers. Users can extend its abilities by installing plug-ins written for the Eclipse Platform, such as development toolkits for other programming languages, and can write and contribute their own plug-in modules.



To start read RDFa file with Jena using SPARQL query:

First step:

Create new project in eclipse→file→new→ java project

Will appear windows:

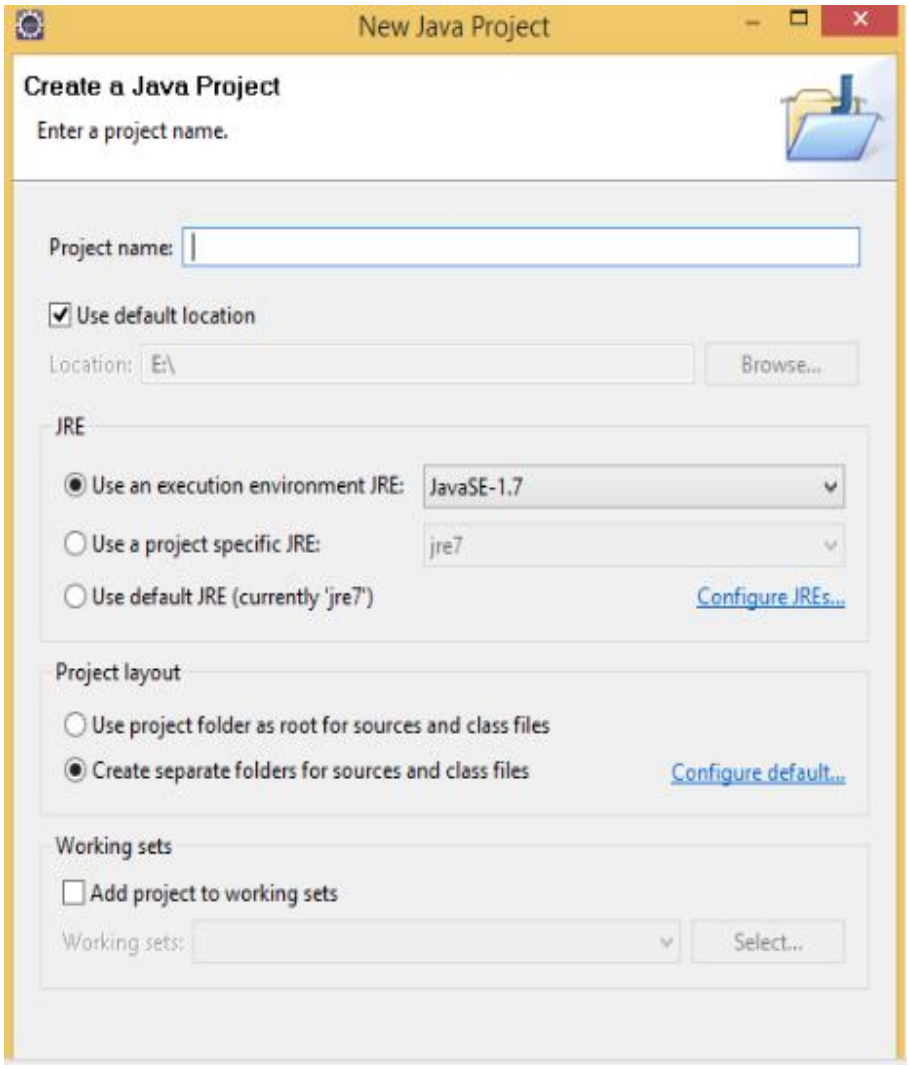


Figure ٦.٣.٤ Create Java Class

Second step:

Add new class to project package as:

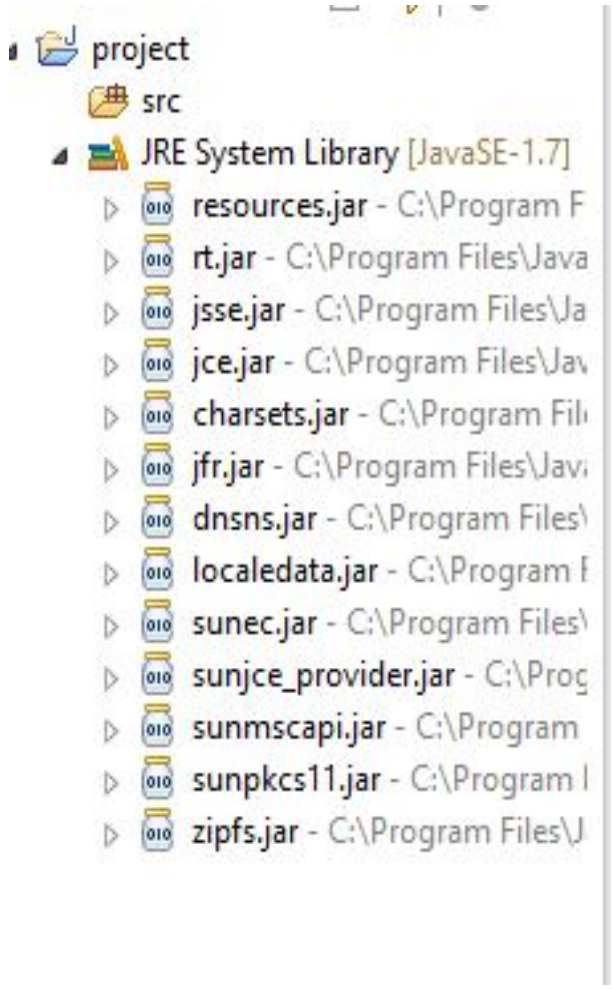
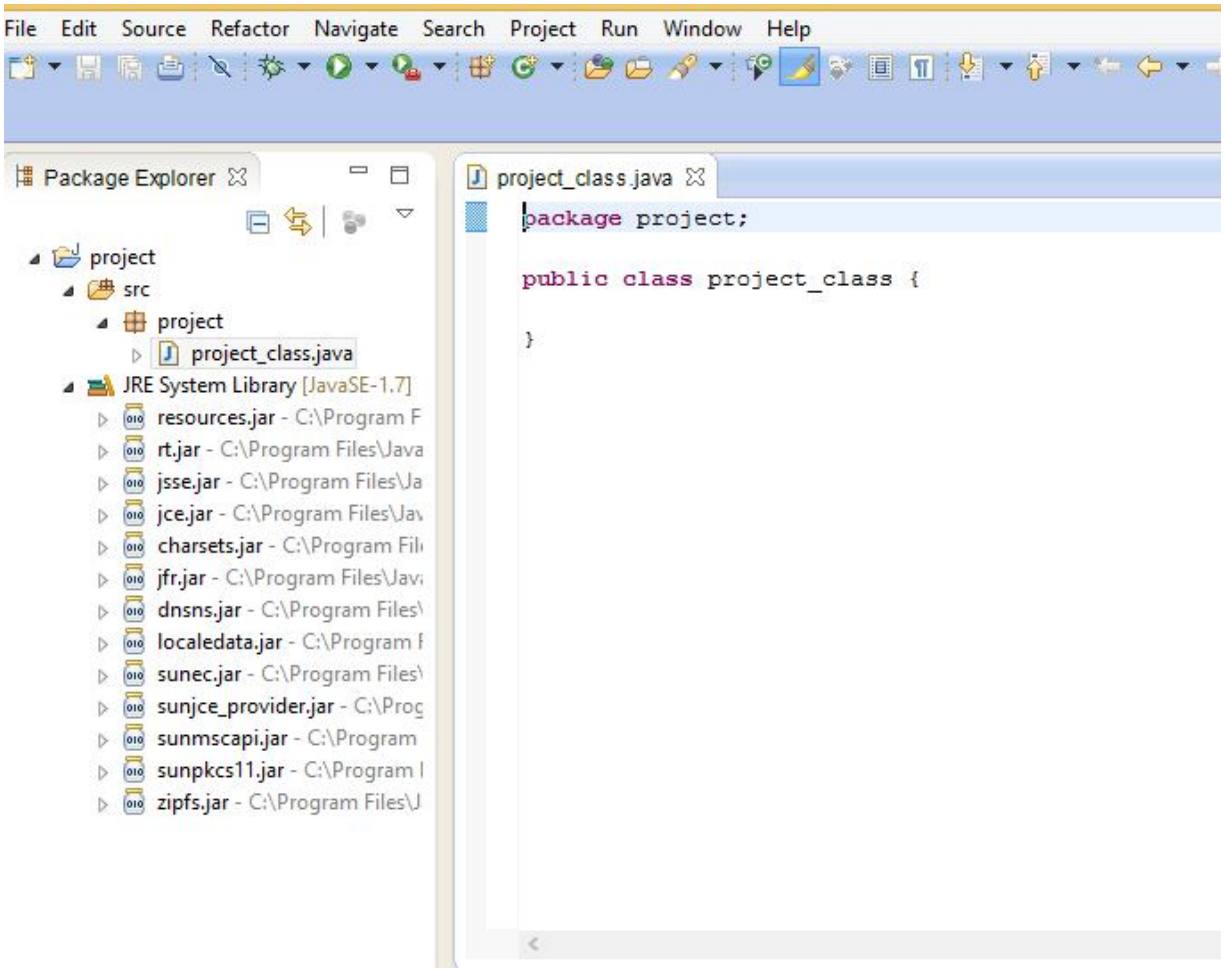
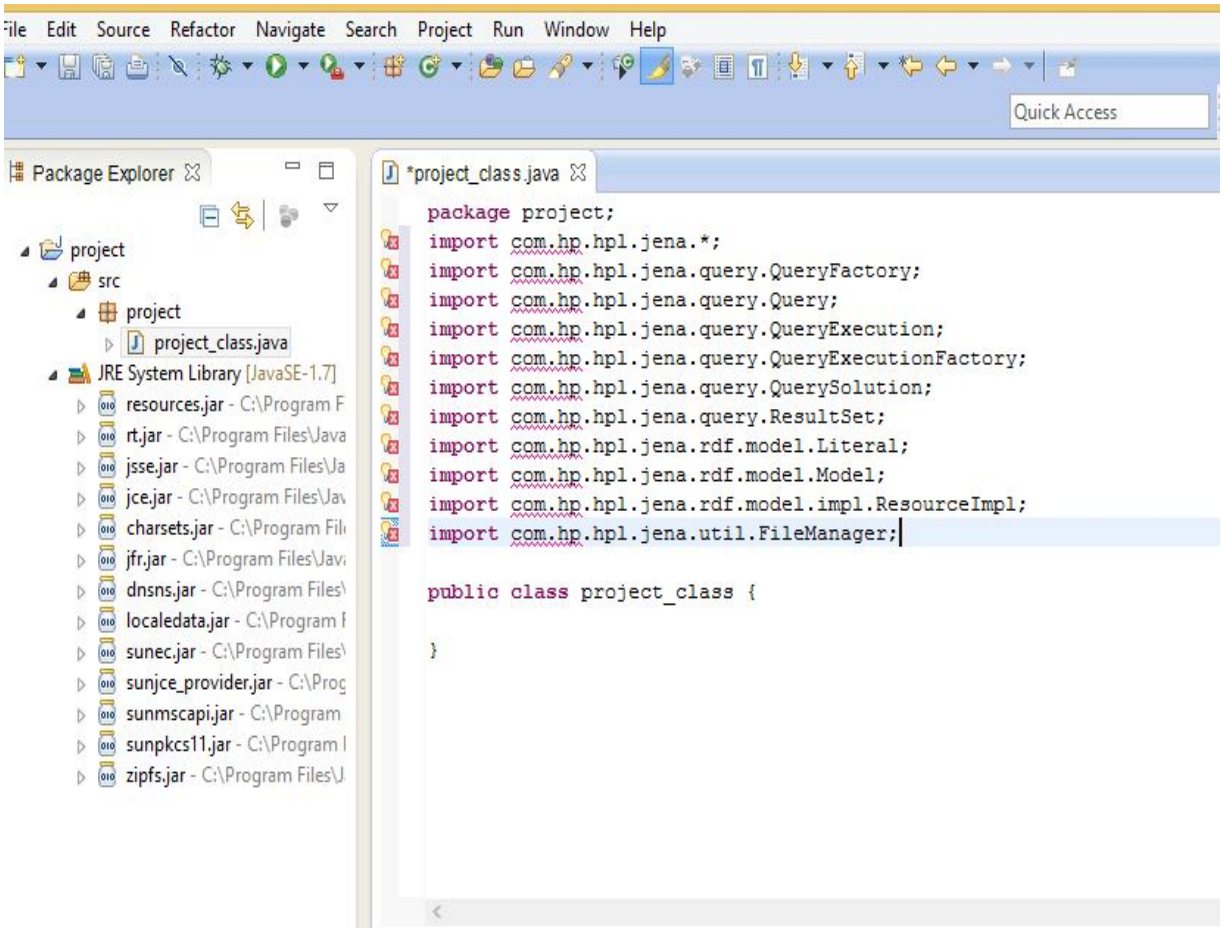


Figure ٦.٣.٤ Create Class

Click on project →new →class



After that add imports which help me to read RDFA file



We look that project cannot know that so we add apache Jena which identify that

Click on project →build bath→user Library

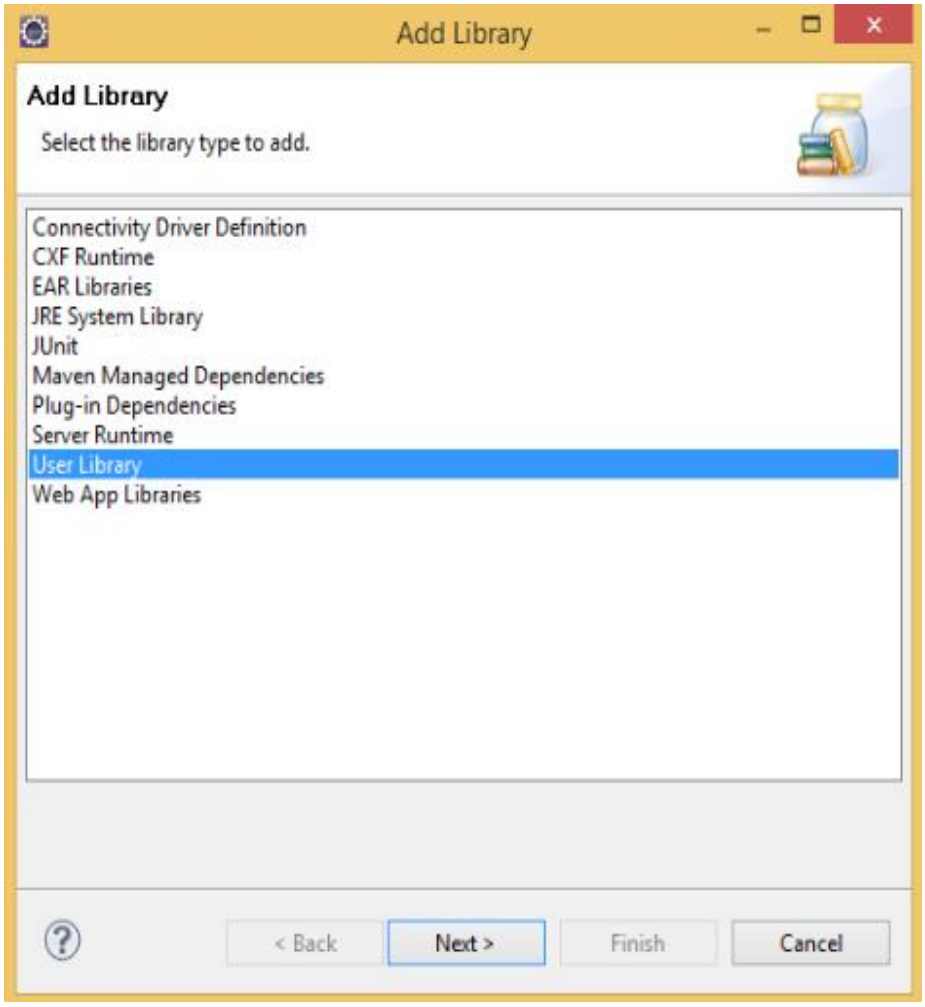


Figure ٦.٣.٤ Add Library

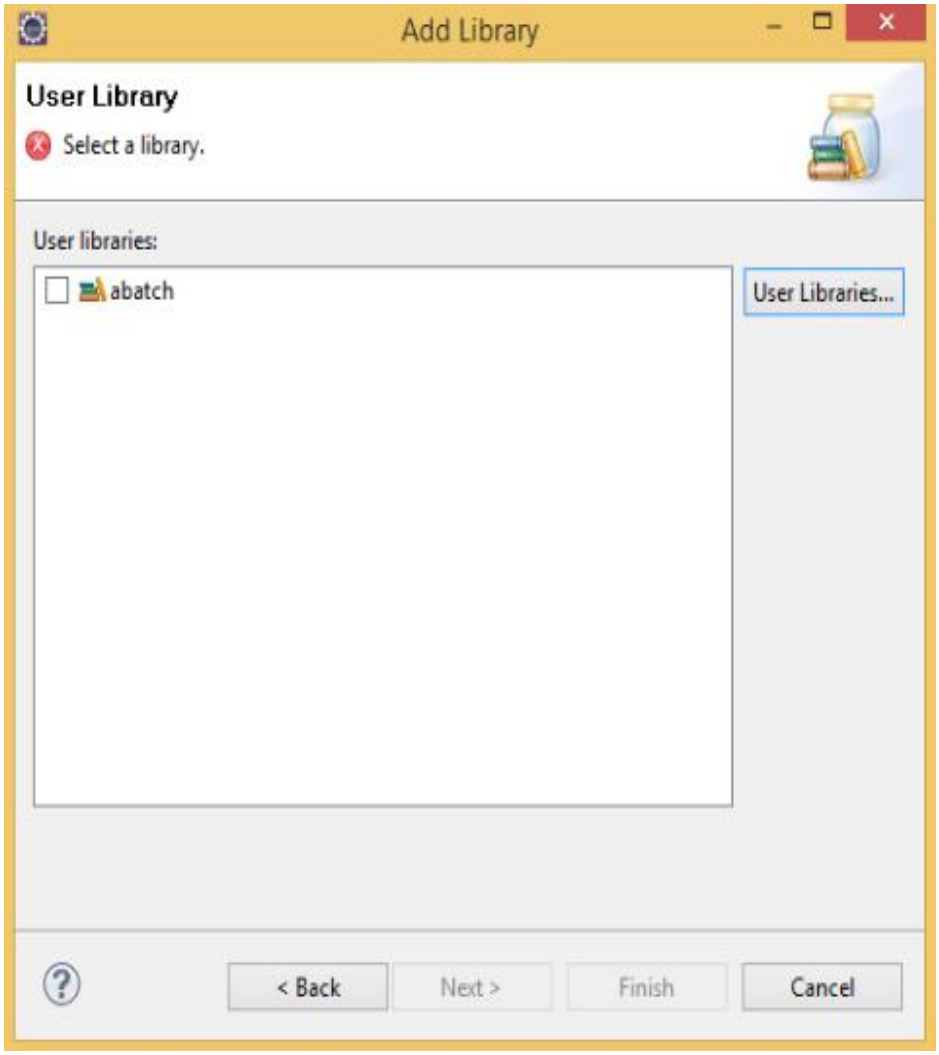
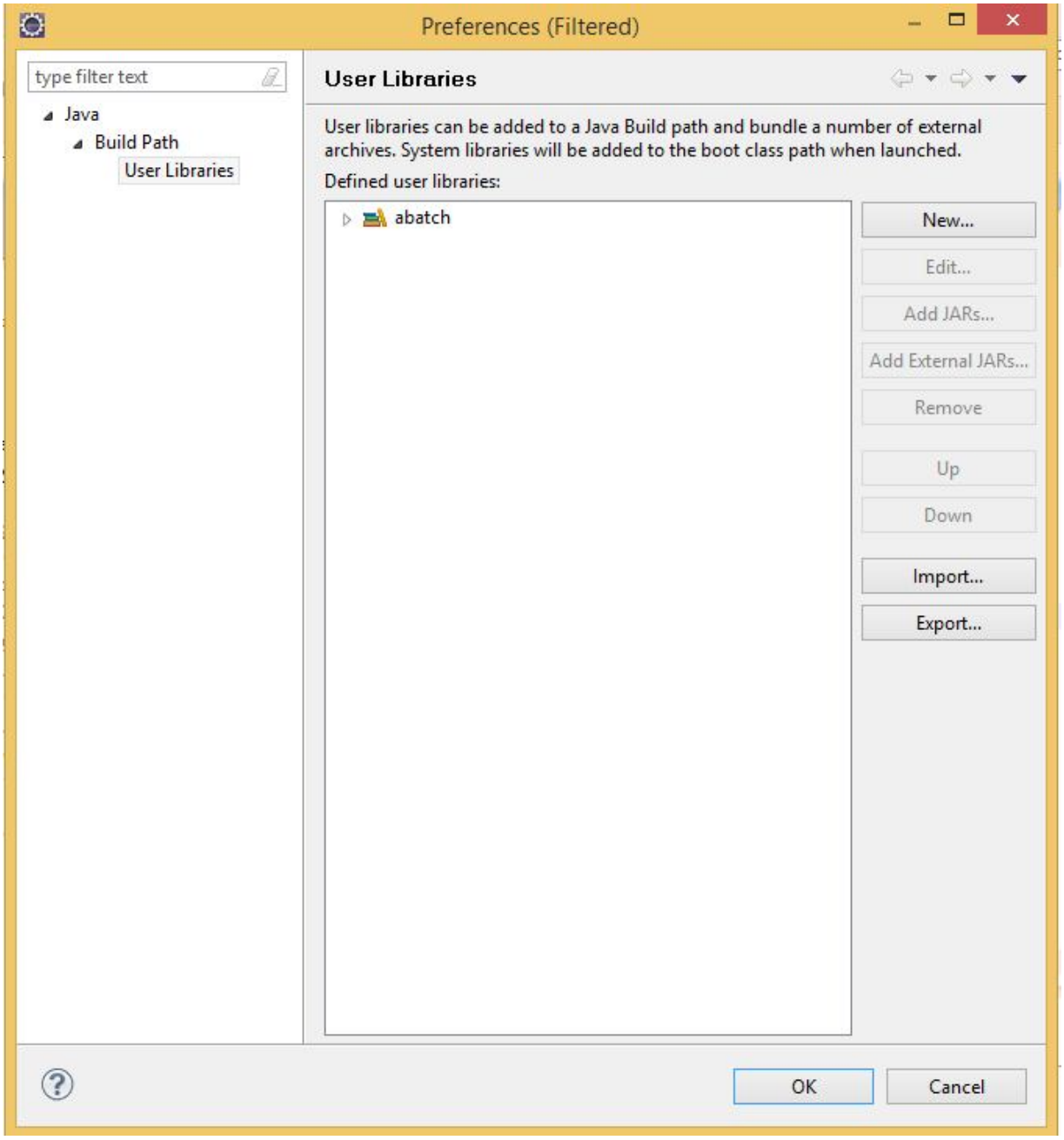


Figure ٦.٣.٤ Add Library

That create new library include lib which I want to add it


Preferences (Filtered)
type filter text
Java
Build Path
User Libraries
User Libraries
User libraries can be added to a Java Build path and bundle a number of external archives. System libraries will be added to the boot class path when launched.
Defined user libraries:
abatch
New...
Edit...
Add JARs...
Add External JARs...
Remove
Up
Down
Import...
Export...
OK
Cancel

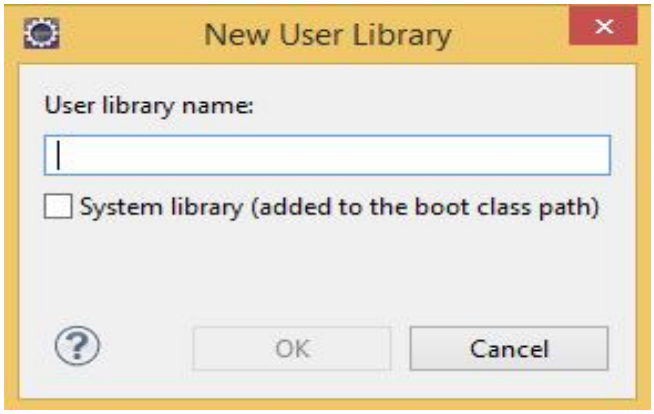


Figure ٦.٣.٤ add library

Add External jars to that file

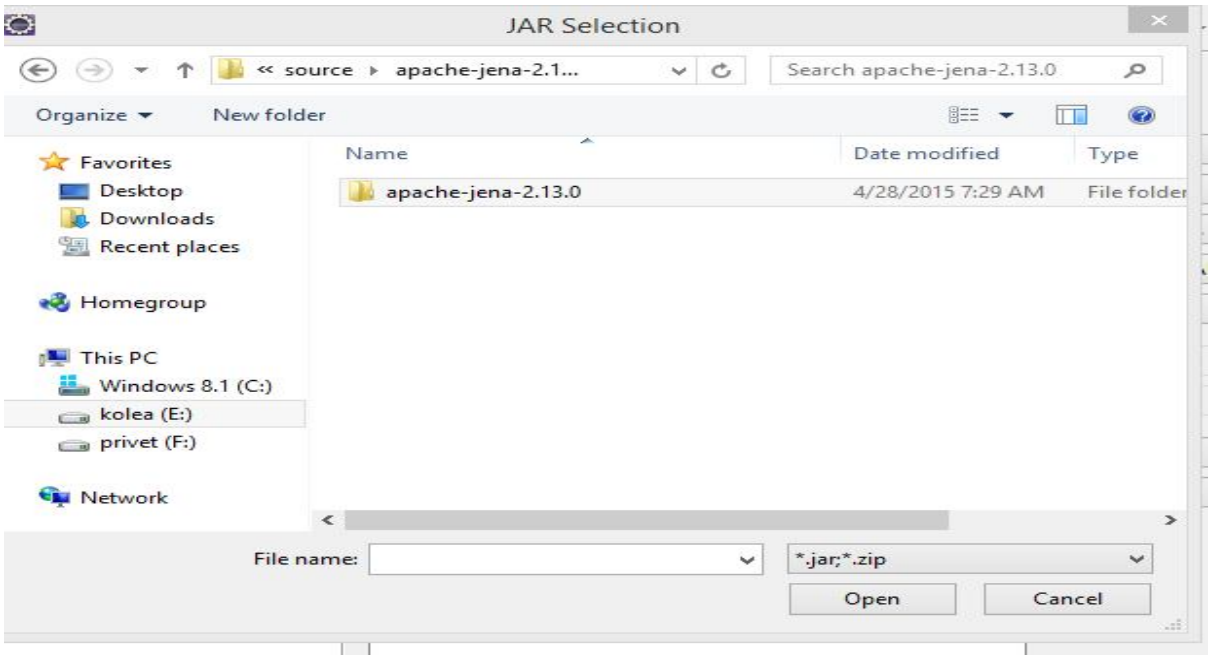


Figure ٦.٣.٤ Select Jar File

This apache –Jena include lib which we need

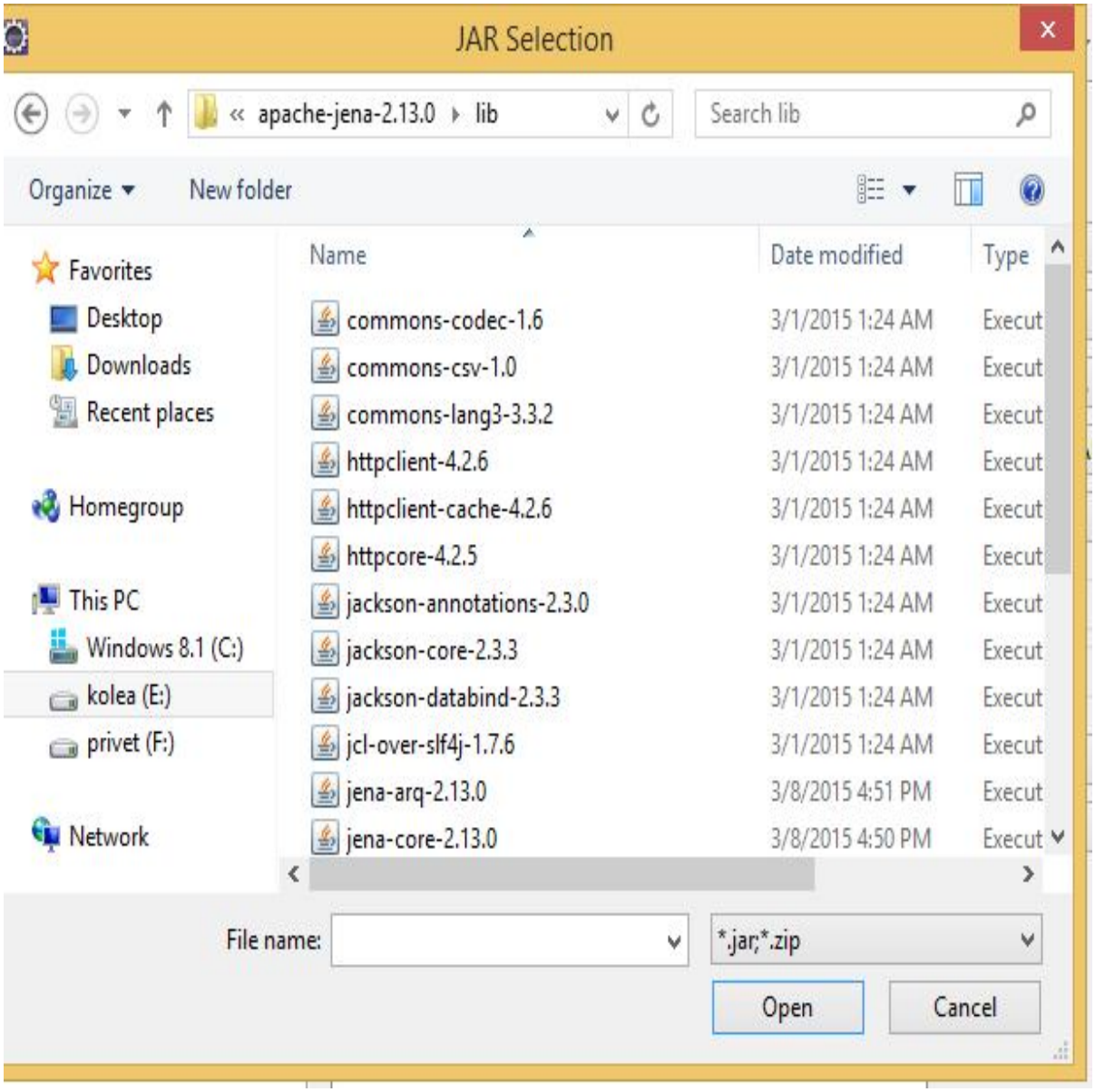


Figure ٦.٣.٤ Select Jar File

Select all and add that to my file

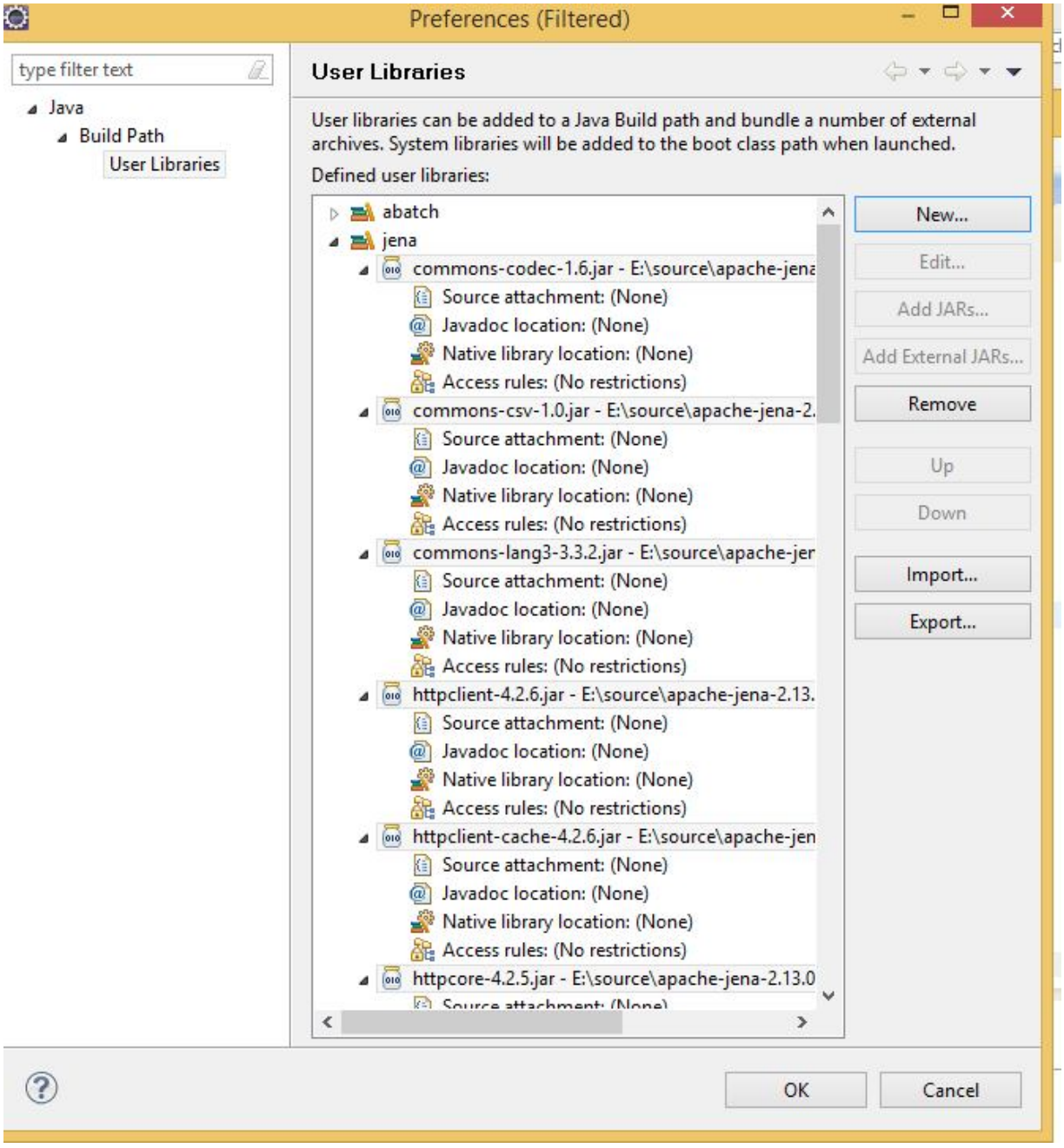


Figure ٦.٣.٤ Add Library

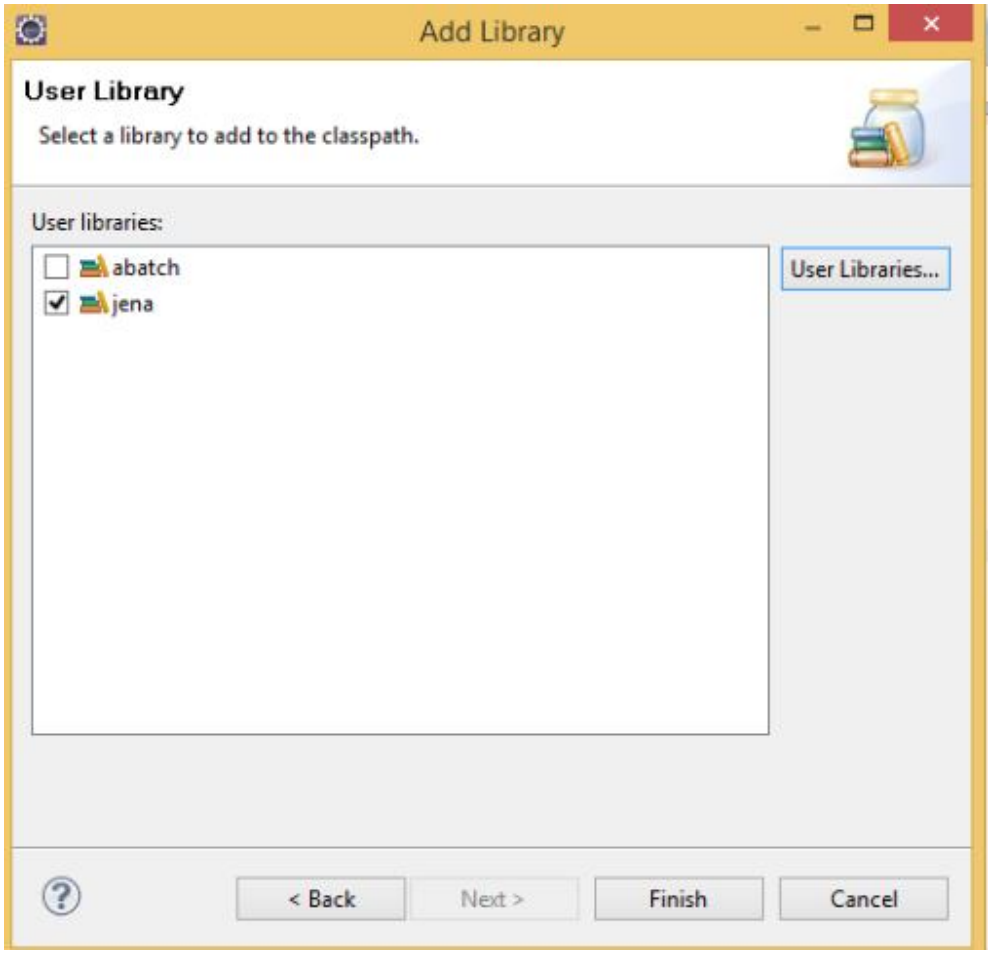
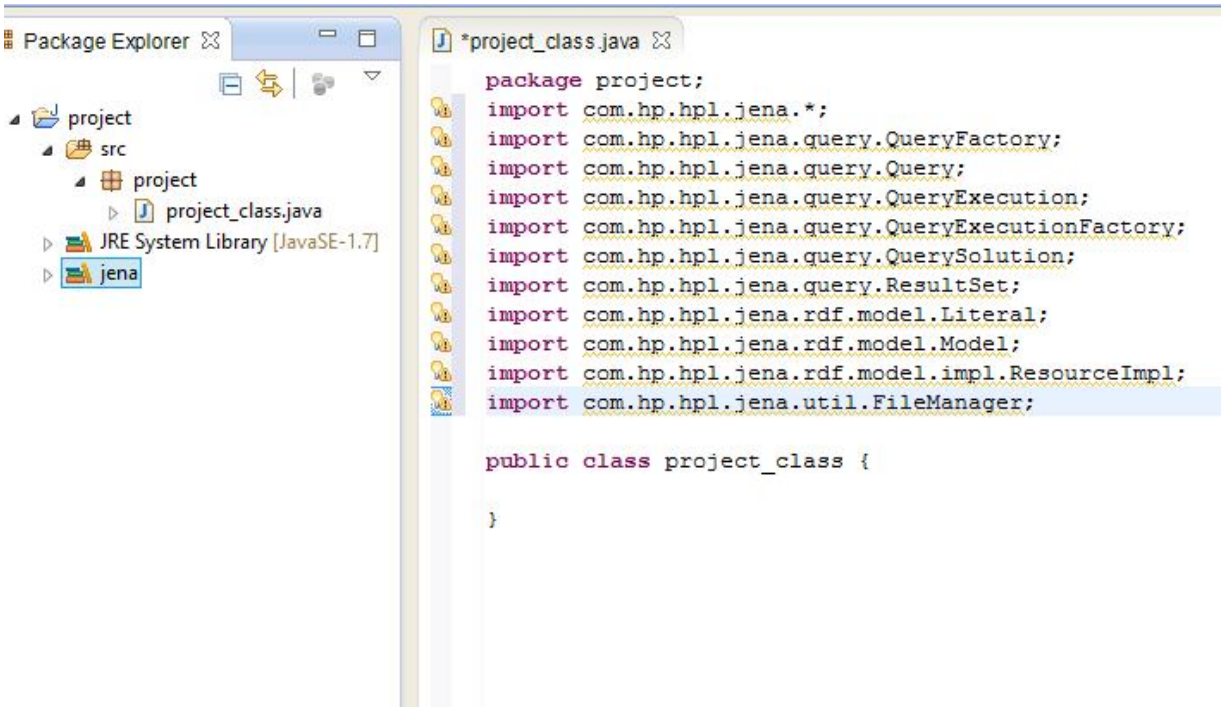


Figure ٦.٣.٤ Add Library

Select file and click finish

This is result that my project can read& identify library which can use it



٦.٤ SPARQL Query

After that we begin connect and use SPARQL query :

SPARQL (pronounced "sparkle", a recursive acronym for SPARQL Protocol and RDF Query Language) is an RDF query language, that is, a semanticquery language for databases, able to retrieve and manipulate data stored in Resource Description Framework (RDF) format. It was made a standard by the RDF Data Access Working Group (DAWG) of the World Wide Web Consortium, and is recognized as one of the key technologies of the semantic web. On ١٥ January ٢٠٠٨, SPARQL ١.٠ became an official W٣C Recommendation. And SPARQL ١.١ in March, ٢٠١٣

SPARQL allows for a query to consist of triple patterns, conjunctions, disjunctions, and optional patterns.

SPARQL allows users to write queries against data that can loosely be called "key-value" data or, more specifically, data that follows the RDF specification of the W٣C. The entire database is thus a set of "subject-predicate-object" triples. This is analogous to some No SQL databases' usage of the term "document-key-value", such as MongoDB.

RDF data can also be considered in SQLrelational database terms as a table with three columns – the subject column, the predicate column and the object column. Unlike relational databases, the object column is heterogeneous, the per-cell data type is usually implied (or specified in the ontology) by the predicate value. Alternately, again comparing to SQL relational, all of the triples for a given subject could be represented as a row, with the subject being the primary key and each possible predicate being a column and the object is the value in the cell.

However, SPARQL/RDF becomes easier and more powerful for columns that could contain multiple values (like "children"), and where the column itself could be a joinable variable in the query, rather than directly specified.

Query forms

In the case of queries that read data from the database, the SPARQL language specifies four different query variations for different purposes.

SELECT Query

Used to extract raw values from a SPARQL endpoint, the results are returned in a table format.

CONSTRUCT Query

Used to extract information from the SPARQL endpoint and transform the results into valid RDF.

ASK query

Used to provide a simple True/False result for a query on a SPARQL endpoint.

DESCRIBE query

Used to extract an RDF graph from the SPARQL endpoint, the contents of which is left to the endpoint to decide based on what the maintainer deems as useful information.

Each of these query forms takes a WHERE block to restrict the query although in the case of the DESCRIBE query the WHERE is optional.

SPARQL ١.١ specifies a language for updating the database with several new query forms.

Add RDFA file to our project:

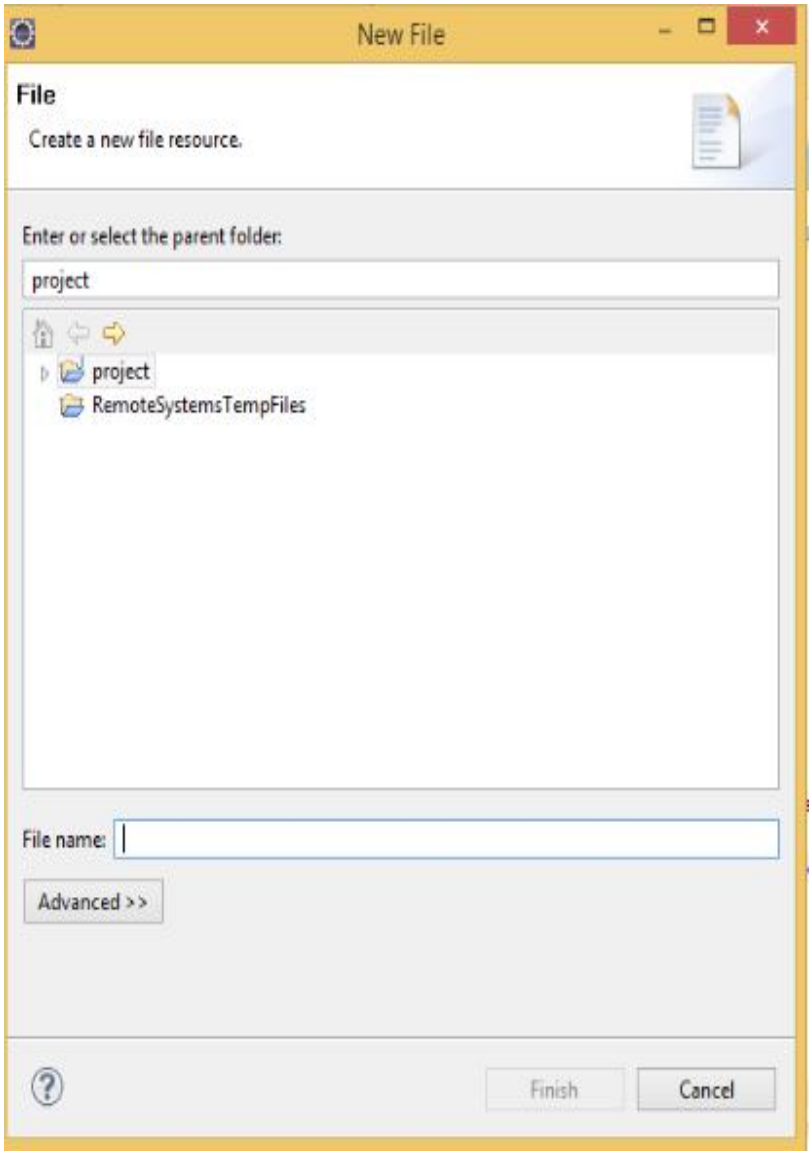


Figure ٦.٤ add RDFA

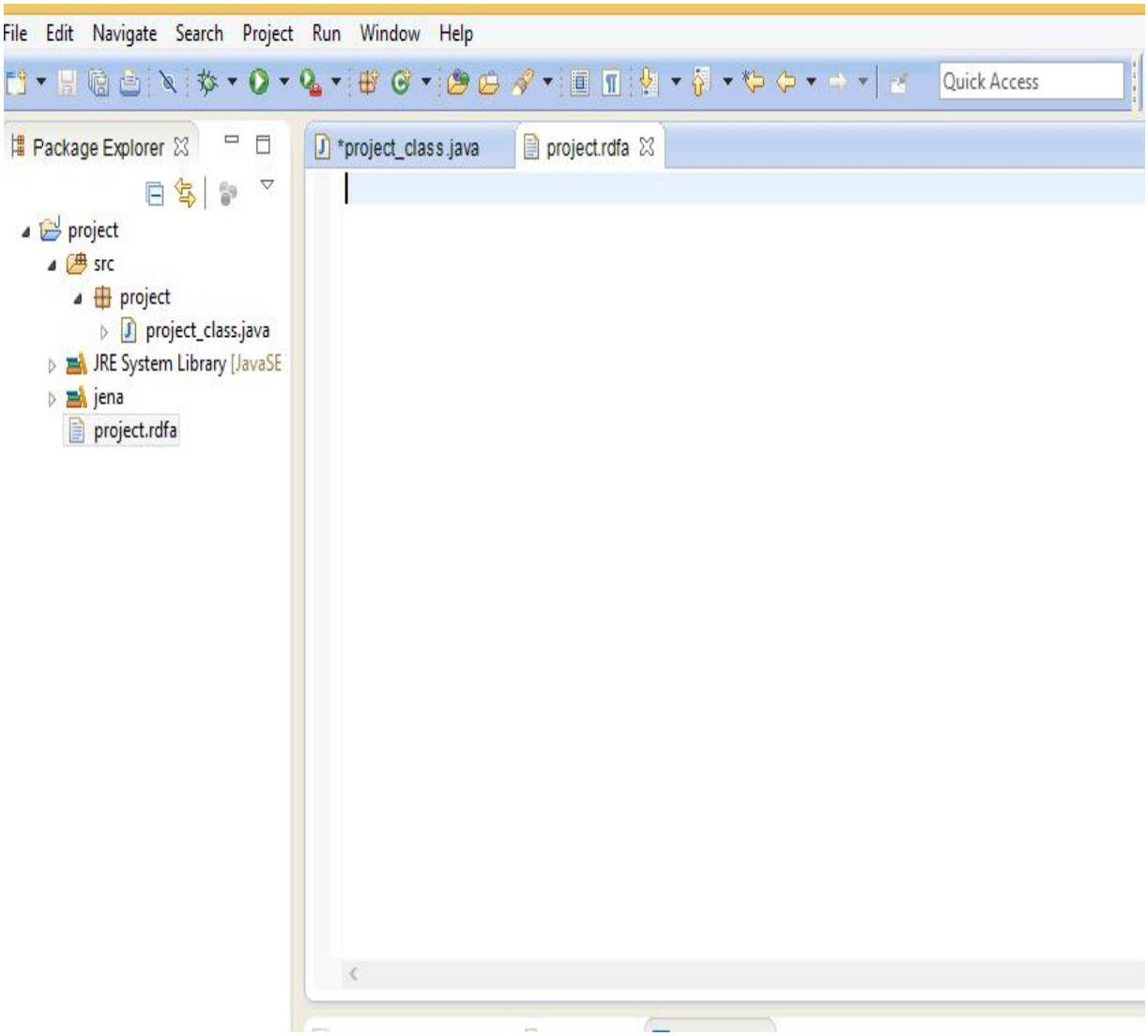


Figure ٦.٤ add RDFA

- Add content my ontology to RDFA file.

Chapter 7

Case Study

٧.١ Introduction

٧.٢ Final result

After working the System Analysis of Public Opinion we reach to good result that will help in opinion classification which plays an important role in our country especially in political field. From this classification political experts can determine problems that face people and help them to reach to good decision.

٧.٢ Final Result

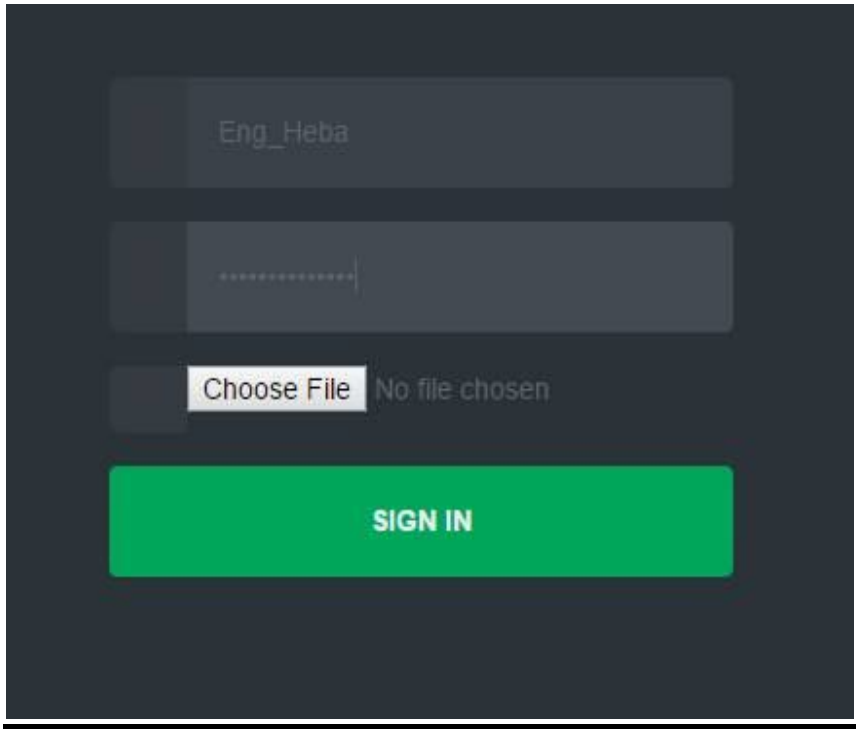


Figure ٧.١ Login to web site

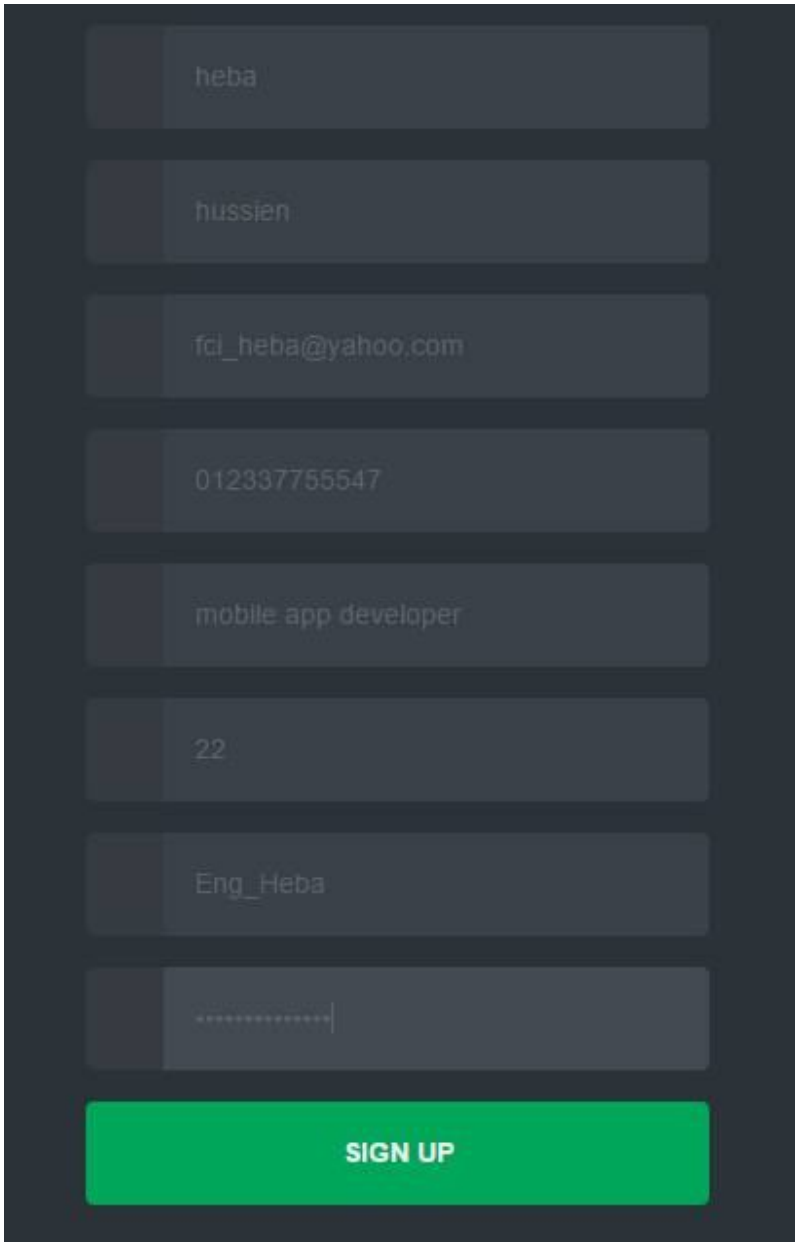


Figure ٧.٢ Add people to system

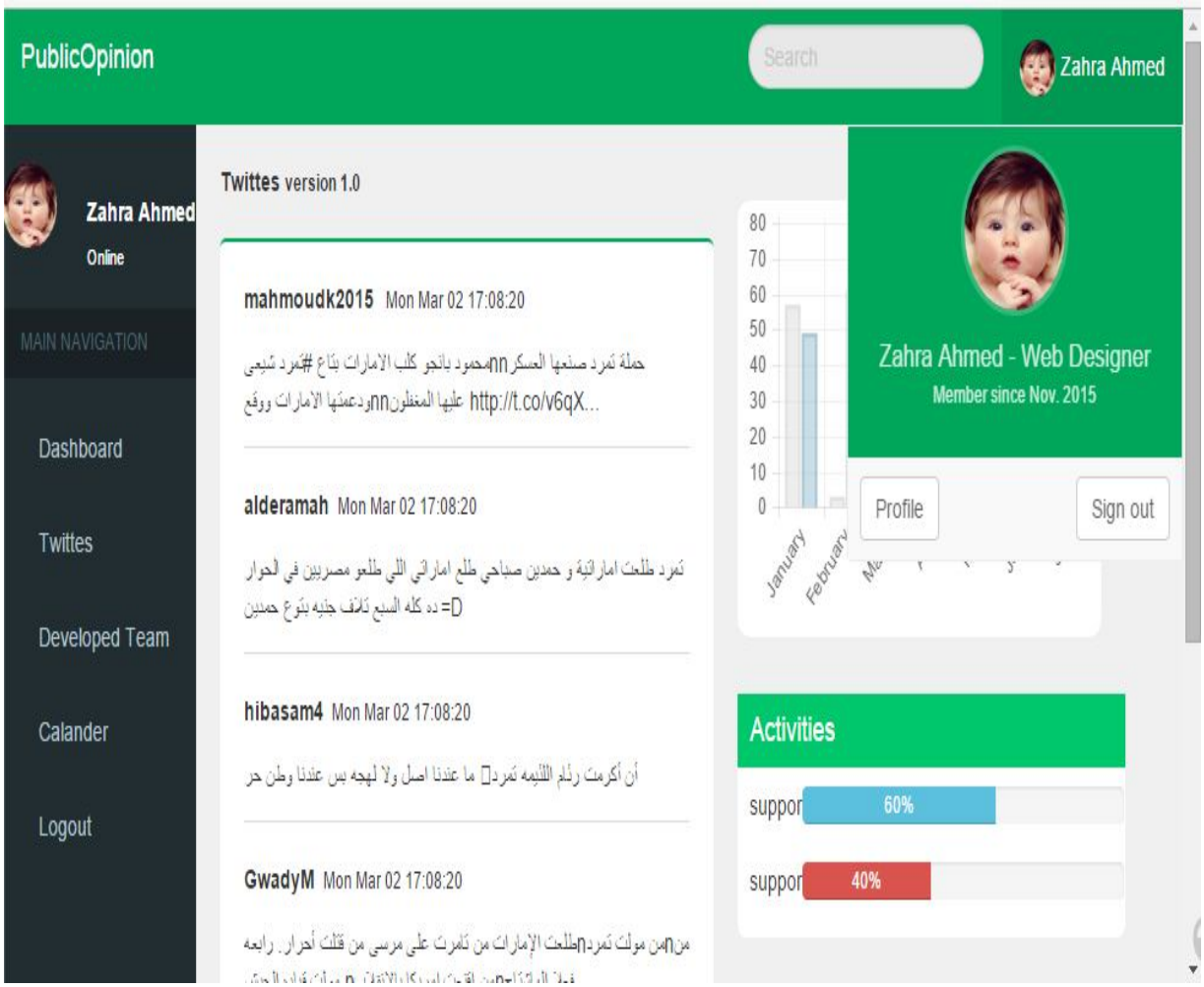


Figure ٧.٢ final result

Chapter 8

Difficulties & Future Work

٨.١ Introduction

٨.٢ Difficulties That Faced Us

٨.٣ Future Work

٨.٤ Conclusion

To develop System Analysis of Public Opinion we faced many problems that disabled our work. These problems cause some limitations so we decided that we will try to solve these problems in the future work.

٨.٢ Difficulties

We want to make application work on all social media but we found difficulties.

١. We take long time to find a tool that provide data that we will use it then in mining to classify people into supporter and opponent.
٢. we work on Arabic that inherentlydifficult ,many difficulties face us in mining as there are not efficient stemmer in Arabic
٣. We first wanted to work on facebook data but it not available so we decided to use twitter.
٤. Semantic was difficult to applied as there were many problems faced us such as to make rules of positive opinion and negative opinion.
٥. Limited time of work.

٨.٣ Future Work

In the future plan of our time we will work on:

١. Big Data with cloud as we will have a lot of data so big data will be suitable.
٢. Using data from other social media like Facebook to make our system more efficient and accurate.

٣.We hope to make efficient website with RDFA.

٨.٤ Conclusion

This chapter includes difficulties that faced our team during project implementation and future work belongs to this project.

References

Chapter ٢:

[٢.٢]: http://www.aucegypt.edu/gapp/cairoreview/pages/articledetails.aspx?aid=٨٩

[٢.٣]: http://www.computerhope.com/jargon/s/socinetw.htm

https://en.wikipedia.org/wiki/Twitter#Logo

https://en.wikipedia.org/wiki/Twitter_Revolution

Chapter ٣:

[٣.٢]: http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=٥٢٣٣٢٠&url=http%٣A%٢F%٢Fieeexplore.ieee.org%٢Fxpls%٢Fabs_all.jsp%٣Farnumber%٣D٥٢٣٣٢٠

[٣.٣]:http://ec.europa.eu/public_opinion/cf/index_en.cfm

http://ec.europa.eu/public_opinion/cf/step١.cfm

[٣.٤]: http://www.scientific.net/AMR.٩٢٦-٩٣٠.٢٢٣٣

[٣.٥]: http://www.kexion.com/en/fxxt.html

Chapter ٤:

[٤.١.١]: https://en.wikipedia.org/wiki/Systems_development_life_cycle

[٤.١.٢]: https://airbrake.io/blog/insight/what-is-system-development-life-cycle

References

[٤.٢]: http://en.wikipedia.org/wiki/planning#Overview

[٤.٣]: http://en.wikipedia.org/wiki/System_Development_Life_Cycle#System_analysis

[٤.٥]: http://databasemanagement.wikia.com/wiki/SDLC#Implementation

http://en.wikipedia.org/wiki/Software_develoment_process#Implementation.٢C_testing_and_documentation

[٤.٦]: http://itknowledgeexchange.techtarget.com/quality-assurance/sdlc-vii-deployment-operations-maintenance-and-disposal/

http://en.wikipedia.org/wiki/Maintenance_repair_and_operations

Chapter ٥:

http://www.anderson.ucla.edu/faculty/jason.frand/teacher/technologies/palace/datamining.htm

Chapter ٦:

[٦.١]: https://en.wikipedia.org/wiki/Semantic_Web

[٦.٢]: http://protege.stanford.edu

https://en.wikipedia.org/wiki/Prot%C٣%A٩g%C٣%A٩_(software)

Glossary

- Use Case Diagram: Use cases represent system functionality from the user's perspective

  Use Case diagrams describe who will use the system and in what ways the user expects to interact with the system.

  Use Case diagrams represent the interactions between use cases and actors

- System Sequence Diagram: A Sequence Diagrams describe how objects interact with each other via messages in the execution of a use case or operation.

  A sequence Diagram is an interaction diagram that emphasizes the time ordering of messages (ordered by time).

- System Collaboration Diagram: Collaboration Diagram is also called a communication Diagram.

  Numbers are used to show the sequence of messages, the number beside the messages are called sequence numbers.

- Class Diagram: Class Diagrams describe the system’s object structure.

  Classes Diagrams show object classes that the system is composed of as well as the relationships between those object classes.

Glossary

Class Diagrams describe the static structure of a system.

- Activity Diagram: An Activity Diagram is used to describe the sequential flow of activities of a use case (flow of functionality) in a system.

  An Activity Diagram is a flow chart showing the flow of activities through the system.

  The activity diagram is similar to state diagram because activities are state of doing something.

- World Wide Web Consortium (W٣C): is an international community where Member organizations, a full-time staff, and the public work together to develop Web standards. Led by Web inventor Tim Berners-Lee and CEO Jeffrey Jaffe, W٣C's mission is to lead the Web to its full potential. Contact W٣C for more information.

- Resource Description Framework (RDF): RDF is a standard model for data interchange on the Web. RDF has features that facilitate data merging even if the underlying schemas differ, and it specifically supports the evolution of schemas over time without requiring all the data consumers to be changed.

Glossary

RDF extends the linking structure of the Web to use URIs to name the relationship between things as well as the two ends of the link (this is usually referred to as a "triple"). Using this simple model, it allows structured and semi-structured data to be mixed, exposed, and shared across different applications.

This linking structure forms a directed, labeled graph, where the edges represent the named link between two resources, represented by the graph nodes. This graph view is the easiest possible mental model for RDF and is often used in easy-to-understand visual explanations.

- SPARQL: defines a standard query language and data access protocol for use with the Resource Description Framework (RDF) data model. It works for any data source that can be mapped to RDF. The specification is under development by the RDF Data Access Working Group (DAWG).

- Ontology Web Language (WOL): is a set of markup languageswhich are designed for use by applications that need to process the content of information instead of just presenting information to humans.

  The OWL ontology describes the hierarchical organization of ideas in a domain, in a way that can be parsed and understood by software. OWL has more facilities for expressing meaning and semantics than XML, RDF, and RDF-S, and thus OWL goes beyond these languages in its ability to represent machine interpretable content on the Web.

Glossary

- XML: Extensible Markup Language (XML) is used to describe data. The XML standard is a flexible way to create information formats and electronically share structured data via the public Internet, as well as via corporate networks.

- XML Schema: is a World Wide Web Consortium (W٣C) recommendation that specifies how to formally describe the elements in an Extensible Markup Language (XML) document.

  This description can be used to verify that each item of content in a document adheres to the description of the element in which the content is to be placed. XSD ١.١ became an approved W٣C standard in April ٢٠١٢.

  - System Requirements: Consist of functional and nonfunctional requirements.

Appendix A

Tweets Preparation

```
package classify;
import com.sun.xml.internal.ws.util.StringUtils;
import java.io.*;
import javax.swing.JTextArea;

////////////////////////////////////////////////////////////////
public class Key_words_extractor {
private JTextArea testString        =new JTextArea();
 private JTextArea testInt          =new JTextArea();
    String lines="";
////////////////////////////////////////////////////////////////
public Key_words_extractor(String fileWord,String fileCount){
int j=٠;
       String[]line=new String[٤٠];
       String test="";
 try{
           File myfile          =new File("file.txt");
```

```
            FileReader reader     = new FileReader(myfile);
            BufferedReader Breader=new BufferedReader(reader);
while((test=Breader.readLine())!=null)
            {
               line[j]=test;
               j++;
            }
            reader.close();
            Breader.close();
        } catch (Exception e) {
            System.out.println("Error"+ e.getMessage());
        }


  ////////////////////////////////////////////////////////////////
        try{
        String [] myfileInwords=new String[٢٠٠٠];
for(int x=٠;x<j;x++){
            String fileAsString=line[x];
            x++;
            fileAsString=fileAsString.toLowerCase();
            char[] ch={'١','٢','٣','٤','٥','٦','٧','٨','٩','٠','-
',_,'%',,,.,'@','*','&',')','(',';',"",'?','|','؟','#','n','e','f','g','k','i','j','l','m','o','p','q','r','s','u','v',
'x','y','z','a','b','d','h','t','\\','P','c','w','!'};
```

```
for(inti=٠;i<ch.length;i++){
            fileAsString=fileAsString.replace(ch[i], ' ');}
            myfileInwords=fileAsString.split("");
            fileAsString=ChangeFiletoString("stopwords.txt");
            String [] stopWordsInwords=fileAsString.split("");

 ////////////////////////////////////////////////////////////////////
        String[] wordsForStem=new String[٢٠٠٠];
int numOfwords=٠;
for(int t=٠;t<myfileInwords.length;t++)
for(int b=٠;b<stopWordsInwords.length;b++)
if(myfileInwords[t].equals(stopWordsInwords[b]))
                    myfileInwords[t]="*";
 for(int u=٠;u<myfileInwords.length;u++)
        {if(myfileInwords[u].equals("*"))
 continue;
else
         {
         wordsForStem[numOfwords]=myfileInwords[u];
         numOfwords++;}}

//////////////////////////////////////////////////////////////
    String Arabicfile="";
```

```
for(int t=٠;t<numOfwords;t++)
    Arabicfile+=wordsForStem[t]+"\n";
    File keyWordsArabic=new File("proj.txt");
    FileWriter writerArabic=new FileWriter(keyWordsArabic);
testString.setText(Arabicfile);
testString.write(writerArabic);
    String myfile="proj.txt";
    String afterstem="afterstem.txt";
    ArabicStemmer arabicStemmer = new ArabicStemmer ( myfile,afterstem);
    myfile=ChangeFiletoString(afterstem);
    String[] ArabicwordsTostemin=myfile.split("");
    //--- display result into files to reuse it in the next step
for(int t=٠;t<ArabicwordsTostemin.length;t++)
lines+=ArabicwordsTostemin[t]+" ";
 lines+="\n";
        }
        File keyWords=new File("myFile.txt");
        FileWriter writer=new FileWriter(keyWords);
testString.setText(lines);
testString.write(writer);
           writer.close();
```

```
            }catch(Exception ee){
            System.err.println("Error"+ ee.getMessage());
           }
    }
public String ChangeFiletoString(String fileName)
    {
    FileReader reader = null;
    String fileContent="";
    String test="";
try{
        File myfile         =new File(fileName);
        reader              = new FileReader(myfile);
        BufferedReader Breader=new BufferedReader(reader);
while((test=Breader.readLine())!=null)
            fileContent+=test+" ";
        reader.close();
        Breader.close();
    } catch (Exception e) {
        System.out.println("Error"+ e.getMessage());
    }
return fileContent;
    }
}
```

ملخص

١)نظرة عامة :

- يعد الويب الدلالى من اهم نتائج تكنولوجيا المعلومات فى هذا العصر . ولقد تم اقتراح الويب الدلالي من قبل مدير منظمة W٣Cتيم برنرز لي كوسيط عالمي لتبادل المعلومات والمعرفة البشرية

.

- الويب الدلالى هو ثورة جديدة في عالم الويب حيث تصبح المعلومات والبياناتقابلة للمعالجة منطقياً من قِبل برامج الحاسوب بحيث تتحول تلك المعلومات والبيانات إلى **شبكة بيانات ذات معنى**، أي أنه يمكن للبرامج الحاسوبية الخاصة أن تعرف ماذا تعني هذه البيانات.

-الويب الدلالي هو الحل الطبيعي لمشاكل الويب، وهو الطريقة التي لا بد منها لجعل إيراد المعلومات الهائل على الويب مفيداً ومستجيباً للبشر؛ حيث لا نستطيع بعد الآن أن نعرض العقل البشري لهذه الكمية الهائلة من البيانات وأن نتوقع منه أن يتعامل معها و يحللها ويستخرج ويركب المعرفة منها.بدلاً من ذلك، يجب أن نكون قادرين على أن نعالج المعلومات أوتوماتيكياً ونقدمها للمستخدم منمذجة ومرتبة لغرض معين، والويب الدلالي يسهل علينا هذه المهمة فهو يصف العلاقات بين الأشياء وخصائصها عن طريق استخدام مجموعة من الأدوات التي تساعد في تحقيق ذلك.

- يهدف البحث فى الويب الدلالى إلى الإجابة عن أمرين أساسيين هما:

- كيف نستطيع أن نمثل المعرفة بطريقة تجعلها قابلة للفهم والمعالجة أوتوماتيكياً من قبل الحواسيب؟
- عندما نحصل على هذه المعلومات؛ كيف يمكننا أن نستفيد منها في التطبيقات الواقعية العملية؟

تطورت وسائل الاتصال بين البشر حديثاً بصورة سريعة ، فبعد أن كان الاتصال قديماً يتم عن طريق الحمام الزاجل والرسل ، تطور الأمر ليصل إلى البريد الورقي والهاتف وغيره حتى وصلنا إلى قمة التطور في طرق التواصل والاتصال حيث ظهر الهاتف الخليوي والشبكة العنكبوتية فكان

منها مثلاً موقع الفيسبوك وتويتروقد أتاحت هذه المواقع الإجتماعية للناس فرصة التواصل بوضع الصور والتواصل عن طريق الكتابة والمحادثة الصوتية ، بل وإن هذه المواقع قد أتاحت للناس فرصة التعلم والتعليم ومشاركة ملفات الفيديو المختلفة ومتابعة الأخبار بصورة آنية محدثة دائماً ، فشبكات التواصل الإجتماعي ساهمت ببناء قاعدة علمية ومعرفية وسهلت الوصول إلى المعلومة وجعلتها بمتناول يد الناس .

-لذلك تم بناء نظام لتحليل الرأى العام باستخدام تكنولوجيا الويب الدلالى لتحليل الكميات الهائله من الاراء و وجهات النظر لمستخدمى شبكات التواصل الاجتماعى للمساعده فى توصيل وجهات نظرهم الى المسؤلين داخل الدوله .

٢)تعريف المشكلة:

فى الفترة الاخيرة ظهر العديد من الثورات احتجاجا على بعض الظروف والاوضاع السيئه . تلك الثورات مع كثرتها وتكررها المستمر قد تؤثر على الحالة العامة للبلاد ولاسيما الوضع الاقتصادى وتتيح الفرص للبعض للتدخل فى شؤن البلاد .

٣)الأهداف:

- تحليل الرأى العام ازاء مشكلة معينة.

-مساعده المسؤلين فى التوصل إلى ما يشغل الرأى العام .

- تحديد مدى رضى الشعب عما تفعله الحكومه من افعال وما تتخذه من قرارات .

٤)خطة المشروع :

- استخلاص البيانات من مواقع التواصل الاجتماعى .

- تحليل البيانات باسخدام بعض وسائل التكنولوجيا الحديثه مثل data mining و

Semantic web.

-تحديد التوجه العام للشعب وعرض النتائج على موقع الكترونى .

٥) هيكل الوثيقة:

تتكون الوثيقة من ٨ فصول وهى :

الفصل الاول المقدمه : ويتضمن هذا الفصل فكرة المشروع بصفه عامه والأهداف العائده منه.

الفصل الثانى خلفيه المشروع: ويتضمن هذا الفصل شرح لشبكات التواصل الإجتماعى مثل الفيس بوك و تويتر كما يوضح دورهم فى الثوره المصريه و دوره فى هذا المشروع

.

الفصل الثالث الأعمال المرتبطه بالمشروع : يوضح الأعمال المرتبطه بنظم تحليل الرأى العام كما توضح مدى الأختلاف بين هذا المشروع والمتواجد من قبل.

الفصل الرابع دورة حياة تطوير المشروع : و يتضمن المراحل من التحليل والتصميم و التخطيط والتنفيذ و الصيانه التى يمر بها المشروع ليصل إلى حالته النهائيه واجراء اختبار على النتائج.

الفصل الخامس تنقيب البيانات : يوضح اهمية مجال تنقيب البيانات فى كثير من المجالات و أهميته فى نظام تحليل الرأى العام عبر شبكات التواصل الإجتماعى .

الفصل السادس الويب الدلالي : هو ثورة جديدة في عالم الويب حيث تصبح المعلومات والبياناتقابلة للمعالجة منطقياً من قِبل برامج الحاسوب بحيث تتحول تلك المعلومات والبيانات إلى **شبكة بيانات ذات معنى** ،"أي أنه يمكن للبرامج الحاسوبية الخاصة أن تعرف ماذا تعني هذه البيانات."

ويوضح هذا الفصل اهمية الويب الدلالى بالنسبه لهذا المشروع.

الفصل السابع دراسة الحاله : ويحتوى على دراسة الحاله النهائيه التى وصل إليها المشروع و إجراء الإختبارات عليه ليكون جاهزا للاستخدام .

الفصل الثامن الخاتمه : وتحتوى على تلخيص الوثيقه و تحدد بها الصعوبات التى واجهتنا اثناء تنفيذ المشروع و الأعمال المستقبليه التى من الممكن العمل بها لتطوير الفكره.