Social Media Aggregator

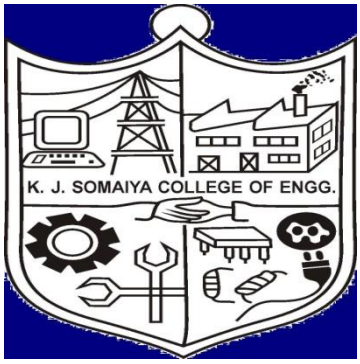
Submitted in Partial fulfilment of the requirements
For the degree of

Bachelor of Engineering
By

Rhythm Shah - 1111071
Riya Patni - 1111074
Vivek Patani - 1111106
Vruksha Shah - 1111107

Under the Guidance of

Prof. Nirmala Shinde



Department of Computer Engineering

K. J. Somaiya College of Engineering, Mumbai

University of Mumbai

2014 – 2015

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

---------- ----------

(Rhythm Shah 1111071) (Signature)

---------- ----------

(Riya Patni 1111074) (Signature)

---------- ----------

(Vivek Patani 1111106) (Signature)

---------- ----------

(Vruksha Shah 1111107) (Signature)

Date:

ACKNOWLEDGEMENT

We take this opportunity to express our profound gratitude and deep regards to our project mentor Prof. Nirmala Shinde for her exemplary guidance, monitoring and constant encouragement throughout the course of the project. The blessing, help and guidance given by her proved a major driving factor for the progress of this project.

We are obliged to staff members of K. J. Somaiya College of Engineering (Computer Engineering Dept.), for the valuable information and support provided by them in their respective fields. We are grateful for their cooperation during the period of our project.

Lastly, we thank the Almighty, our family and friends for their constant encouragement without which this project would not be possible.

Abstract

Police agencies across the globe are increasingly using Online Social Media (OSM) to acquire intelligence and connect with citizens. Developed nations have well thought of strategies to use OSM for policing. However developing nations like India are exploring an evolving OSM as a policing solution.

OSM offers velocity, variety, veracity and large volume of information. Success of police initiatives on OSM to maintain law & order depends both on their understanding of OSM and citizens' acceptance & participation on these platforms.

In this project we aim at identifying the relevant information, categorizing and providing the same to the police. For this we propose using the Android UI and Desktop UI in order to display the information.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Today, with the exponential growth of the web, the amount of news articles generated is huge. It is impossible for anyone to read all the articles at once. Also, not every news article is useful for everyone. So there should be some sort of personalization so that a person gets to see only those articles which may be of relevance to him.

With this intent, we planned to design an application which helps the Mumbai Police view news articles that are of interest to them. For this, we suggested using a focused crawler. A focused crawler is one that attempts to download only those web pages that are relevant to a predefined topic or set of topics, instead of crawling entire WWW. The web is dynamic and it keeps on changing, so it is important to keep track of web pages that change very frequently. In order to determine whether a web page is relevant or not, focused crawler uses various classification techniques.

Thus, we combined the use of focused crawler and developed an application that displays news articles selected by the crawler.

1.1.1 Background

Police system uses a very traditional method of gathering the information. The data collection process is not a very standard process and the police department uses very robust technique of collecting these data.

Different types of papers, forms, documents are collected, and then a rough a database is made with them. Nothing is online. Police department is not well worse with the budding technology and

doesn’t use online social media at all. Thus police is usually unaware about the different society trends.

1.2 Literature Survey

Literature survey includes study of various algorithms that help crawling the web space and collecting relevant results. The whole purpose of literature survey was to decide which approach suits our application the best, by comparing various pros and cons of all the algorithms.

The literature survey also included study of natural language processing needed to carefully design the crawler such that it interprets user query flawlessly and gives most relevant results. However, as per the demand by police, a predefined list of sensitive words was created, which can be edited by the admin.

Chapter 2

Literature Survey

2.1 Algorithms Identified

In order to crawl the web space, various algorithms are available, each with a different approach. For our purpose, we needed an algorithm which is quick, accurate, and which gives only relevant response [14]. After referring various standard IEEE papers, certain algorithms were identified. These are listed as follows:-

1. Breadth First Search Algorithm

2. Depth First Search Algorithm

3. Page Rank Algorithm

4. Genetic Algorithm

5. Naïve Bayes Classification Algorithm

1. Breadth First Search Algorithm

This algorithm aims at uniform search across the neighbour nodes. It starts at the root node and searches all the neighbour nodes at the same level. If the objective is reached, then it is reported as success and the search is terminated. If it is not, it proceeds down to the next level sweeping the search across the neighbour nodes at that level and so on until the objective is reached. When all the nodes are searched, but the objective is not met then it is reported as failure.

Breadth first is well suited for situations where the objective is found on the shallower parts in a deeper tree. It will not perform so well when the branches are so many in a

game tree, especially like chess game and also when all the path leads to the same objective with the same length of the path[7].

2. Depth First Search Algorithm

This powerful technique systematically traverses through the search by starting at the root node and traverses deeper through the child node. If there is more than one child, then priority is given to the left most child and traverse deep until no more child is available. It is backtracked to the next unvisited node and then continues in a similar manner [3].

This algorithm makes sure that all the edges are visited once breadth-wise[4]. It is well suited for search problems, but when the branches are large then this algorithm takes might end up in an infinite loop[5].

3. Page Rank Algorithm

Page rank algorithm determines the importance of the web pages by counting citations or back links to a given page. The page rank of a given page is calculated as

$$\mathrm{PR(A)} = (1 - \mathrm{d}) + \mathrm{d}\left(\frac{\mathrm{PR(T1)}}{\mathrm{C(T1)}} + \ \ldots + \frac{\mathrm{PR(Tn)}}{\mathrm{C(Tn)}}\right)$$

$$\mathrm{PR(A)} = \quad \text{Page Rank of a Website}$$

$$d = damping\ factor$$

$$T1, \ldots, Tn = Links$$

To introduce page belief recommendation mechanism, an algorithm was proposed taking the human factor into consideration, which brought forward a balanced rank algorithm based on PageRank and page belief recommendation which ultimately attaches importance into the subjective needs of the users; so that it can effectively avoid topic drift problems. Tian Chong[6] proposed a new type of algorithm of page ranking by

combining classified tree with static algorithm of PageRank, which enables the classified tree to be constructed according to a large number of users' similar searching results, and can obviously reduce the problem of Theme-Drift, caused by using PageRank only, and problem of outdated web pages and increase the efficiency and effectiveness of search. J.Kleinberg [8] proposed a dynamic page ranking algorithm. Shaojie Qiao [9] proposed a new page rank algorithm based on similarity measure from the vector space mode, called Sim Rank, to score web pages. They proposed a new similarity measure to compute the similarity of pages and apply it to partition a web database into several web social networks (WSNs).

4. Genetic Algorithm

Genetic algorithm is based on biological evolution whereby the fittest offspring is obtained by crossing over of the selection of some best individuals in the population by means of fitness function. In a search algorithm solutions to the problem exists but the technique is to find the best solution within specified time[12] .The genetic algorithm is best suited when the user has literally no or less time to spend in searching a huge database and also very efficient in multimedia results. While almost all conventional methods search from a single point, Genetic Algorithms always operates on a whole population. This contributes much to the robustness of genetic algorithms. It reduces the risk of becoming trapped in a local stationary point .

5. Naïve Bayes classification Algorithm

Naïve Bayes algorithm is based on Probabilistic learning and classification. It assumes that one feature is independent of another . This algorithm proved to be efficient over many other approaches although its simple assumption is not much applicable in realistic cases.[11]

2.2 Comparison of Algorithms

The main objective of the survey was to throw some light on the web crawling algorithms. We also discussed the various search algorithms and the researches related to respective algorithms and their strengths and weaknesses associated. We believe that all of the algorithms surveyed are effective for web search, but the advantages favour more for Breadth First Search with some modifications which are suitable for searching news articles on websites.

2.3 Merits and Demerits of Algorithms

1. Breadth First Search algorithm

Merits:

Performs uniform search across neighbor nodes. One level at a time. Suitable when objective is found on the shallower parts in a deeper tree.

Demerits:

It will not perform so well when the branches are so many in a game tree, especially like chess game and also when all the path leads to the same objective with the same length of the path.

2. Depth First Search algorithm

Merits:

This algorithm makes sure that all the edges are visited once breadth-wise. It is well suited for search problems.

Demerits:

When the branches are large then this algorithm takes might end up in an infinite loop.

3. Page Rank Algorithm

Merits:

Determines the importance of web page by counting citations or backlinks to a given page.

Demerits:

One main disadvantage of PageRank is that it favours older pages. A new page, even a very good one, will not have many links unless it is part of an existing site.

4. Genetic Algorithm

Merits:

This algorithm is more advantageous due to its iterative selection from the population to produce relevant results.

Demerits:

There is no absolute assurance that a genetic algorithm will find a global optimum. It happens very often when the populations have a lot of subjects.

5. Naïve Bayes Algorithm

Merits:

The Naive Bayes classifier's beauty is in its simplicity, computational efficiency, and good classification performance. In fact, it often outperforms more sophisticated classifiers even when the underlying assumption of (conditionally) independent predictors is far from true. This advantage is especially pronounced when the number of predictors is very large.

Demerits:

Three issues should be kept in mind, however. First, the naive Bayes classifier requires a very large number of records to obtain good results. Second, where a predictor category is not present in the training data, naive Bayes assumes that a new record with that category of the predictor has zero probability. This can be a problem if this rare predictor value is important.

Chapter 3

Problem Description and Scope

Social media aggregator is a portal of information collected from various sources such as blogs and news websites so the user need not visit various news boards, creating a personal news space. Android App and Desktop environment are the proposed GUI for the news aggregator. The concept states that the social media aggregator will be available on the android platform in which, the news will be collected via web crawlers from social networks, blogs and news sites. This is being specially designed for the Mumbai Police and is not for Public use.

Our aim is to continuously crawl the related resources, using a focused crawler, and identifying and categorizing the obtained data. The online fetching will be based on latest announcements of the ping server machines.

We are limiting our scope to few important news websites and blogs in order to control the volume of data being crawled and stored in our database.

We aim at:

1. Awareness of various topics trend
2. Gaining the society pulse.
3. Finding Blogger Society Interests.
4. Immediate awareness of the impact of accidents and events on Public opinion.

Chapter 4

Software Project Management Plan (SPMP)

4.1. Introduction

Social Media Aggregator is designed for Mumbai Police with the intention of giving them access to relevant news articles from a set of news websites such that the articles are based on the set of predefined sensitive words.

4.1.1. Project Overview

Police agencies across the globe are increasingly using Online Social Media (OSM) to acquire intelligence and connect with citizens. Developed nations have well thought of strategies to use OSM for policing.

In this project we aim at identifying the relevant information, categorizing and providing the same to the police. For this purpose we are using the Android UI and Desktop UI in order to display the information.

4.1.2. Project Deliverables

- Source code
- Executable code
- Database
- Software design document
- Software test document
- Final Product

4.2. Project Organizations

4.2.1. Software Process Model

For the development of the project we are undertaking the prototyping model, thus dividing the whole workspace in to small deliverables and then merging them thus providing a better application in iterative process.

Prototype Model

Software prototyping is the development approach of activities during software development, the creation of prototypes, i.e., incomplete versions of the software program being developed [25].

The basic principles are:

a) Not a standalone, complete development methodology, but rather an approach to handle selected parts of a larger, more traditional development methodology (i.e. incremental, spiral, or rapid application development (RAD)).

b) Attempts to reduce inherent project risk by breaking a project into smaller segments and providing more ease-of-change during the development process.

c) User is involved throughout the development process, which increases the likelihood of user acceptance of the final implementation.

d) Small-scale mock-ups of the system are developed following an iterative modification process until the prototype evolves to meet the user's requirements.

e) While most prototypes are developed with the expectation that they will be discarded, it is possible in some cases to evolve from prototype to working system.

f) A basic understanding of the fundamental business problem is necessary to avoid solving the wrong problems.

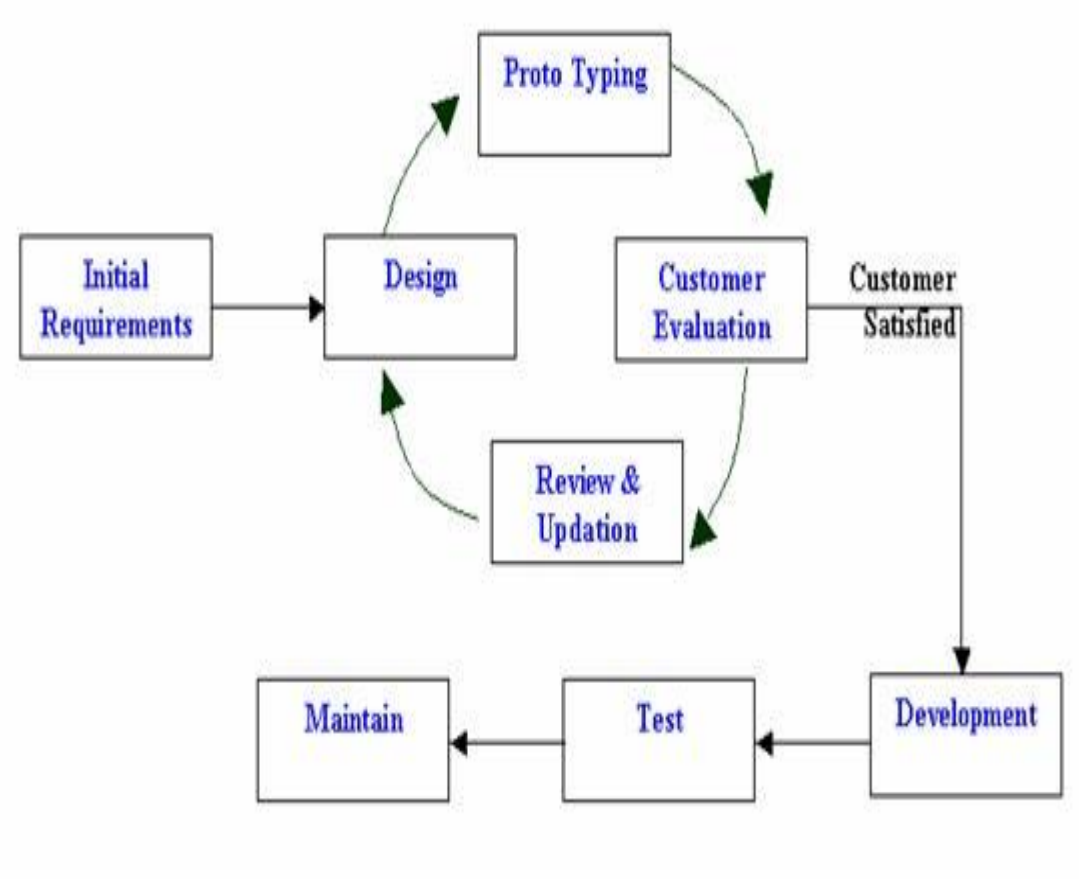


Figure 1 Prototype Model [25]

4.2.2. Tools and Techniques

I. Tools Used

- JCreator to design the Web Crawler, for collection of data.
- Java IDE to design the Android User Interface.
- WAMP (Windows Apache MySQL PHP) to create a virtual environment for the execution of the Crawler and Database.
- Creately (Online Open Source Tool) for UML diagrams [30]

- Notepad++ for Source Code editing.
- Microsoft Word 2013 for report documentation.
- Microsoft PowerPoint 2013 for Presentation.

II. Programming Languages & Techniques Used

- Java - To design the Web Crawler, for collection of data. Also used to design the Android Application with certain imported packets from the Android SDK.
- XML - Used to update data periodically on the Android UI.
- JSON - Used to compare the timestamps on the server with the client device.
- PHP - Used in archiving old information, also used to access data.
- HTML - Used to design the Web UI in association with CSS, JavaScript and PHP.
- SQL -Used in association with Java and HTML to inject data and retrieve data.

4.3. Project Management Plan

4.3.1. Tasks

4.3.1.1. Requirement documentation

4.3.1.1.1. Description

A proper documentation of all the requirements involved in the project. A careful study of all the possible requirements. This includes documenting the network and resource requirements of the aggregator.

4.3.1.1.2. Deliverables and Milestones

Software Requirement Specification will be drafted after the proper requirement study. The project can then be taken in planning and implementation stage.

4.3.1.1.3. Resources Needed

Access to research papers based on working of focused crawler and study of various algorithms in order to understand relevance calculation. Tutorials on implementing Android UI will also be needed.

4.3.1.1.4. Risks and Contingencies

If the requirements are not properly documented, the project progress would drift from the expected performance, rendering a sub-optimal product. Thus, quality of SRS decides quality of the final product. Care should be taken while documenting the SRS.

4.3.1.2. Planning and Design

4.3.1.2.1. Description

Planning involves properly dividing the work among the group members and carefully start working on the advancement of the project. Design involves details on programming languages and environments, packages, application architecture, database design and user interface.

4.3.1.2.2. Deliverables and Milestones

Software Project Management (SPM) and Software Design Document (SDD) will be produced which will then be used for the implementation stage. Based on the guidelines of these documents, development of the project will take place in a prescribed manner.

4.3.1.2.3. Resources Needed

Software Requirement Specification will be used to start the work. Understanding of various design patterns and styles with their advantages and disadvantages will be needed.

4.3.1.2.4. Dependencies and Constraints

To start proper planning and carry out designing of the project, a well formed SRS needed. Better the SRS, better would be the design. The effectiveness of the design will be subject to the complexity of it. Care should be taken, to keep the design simple and allocate sufficient time for each module.

4.3.1.2.5. Risks and Contingencies

Faults in the design will have a snowball, leading to a cascading effect which will deteriorate the overall implementation of the project. A dip in the quality of the design will be reflected in the performance of the application.

4.3.1.3. Implementation and Testing

4.3.1.3.1. Description

This phase involves developing the application and testing the output generated. This is the most important phase wherein the actual application will be developed using the chosen programming language. Various modules will be tested individually followed by a test after integration of all the modules.

4.3.1.3.2. Deliverables and Milestones

Prototype will be generated after each iteration of implementation. Each prototype will be tested and improved. The final product would be delivered when no further improvements will be needed

4.3.1.3.3. Resources Needed

Software Requirement Specification, Software Design Document, Software Project Management Plan are required. Knowledge about programming languages and designing tools is necessary. A list of test cases and test approaches will be needed to test the overall working of the product.

4.3.2. Assignments

Table 1 Responsibility Assignment

Sr No	Tasks	Responsibility
1	Requirement analysis & definition	Vruksha Shah, Riya Patni, Rhythm Shah, Vivek Patani
2	System Requirement Specifications	Vruksha Shah, Rhythm Shah
3	Software Design Document	Riya Patni, Rhythm Shah
4	Implementaion(User Interfaces)	Vivek Patani
5	Implementation (Crawler)	Rhythm Shah, Riya Patni, , Vruksha Shah
6	Software Testing Description	Vivek Patani, Riya Patni

4.3.3. Timetable

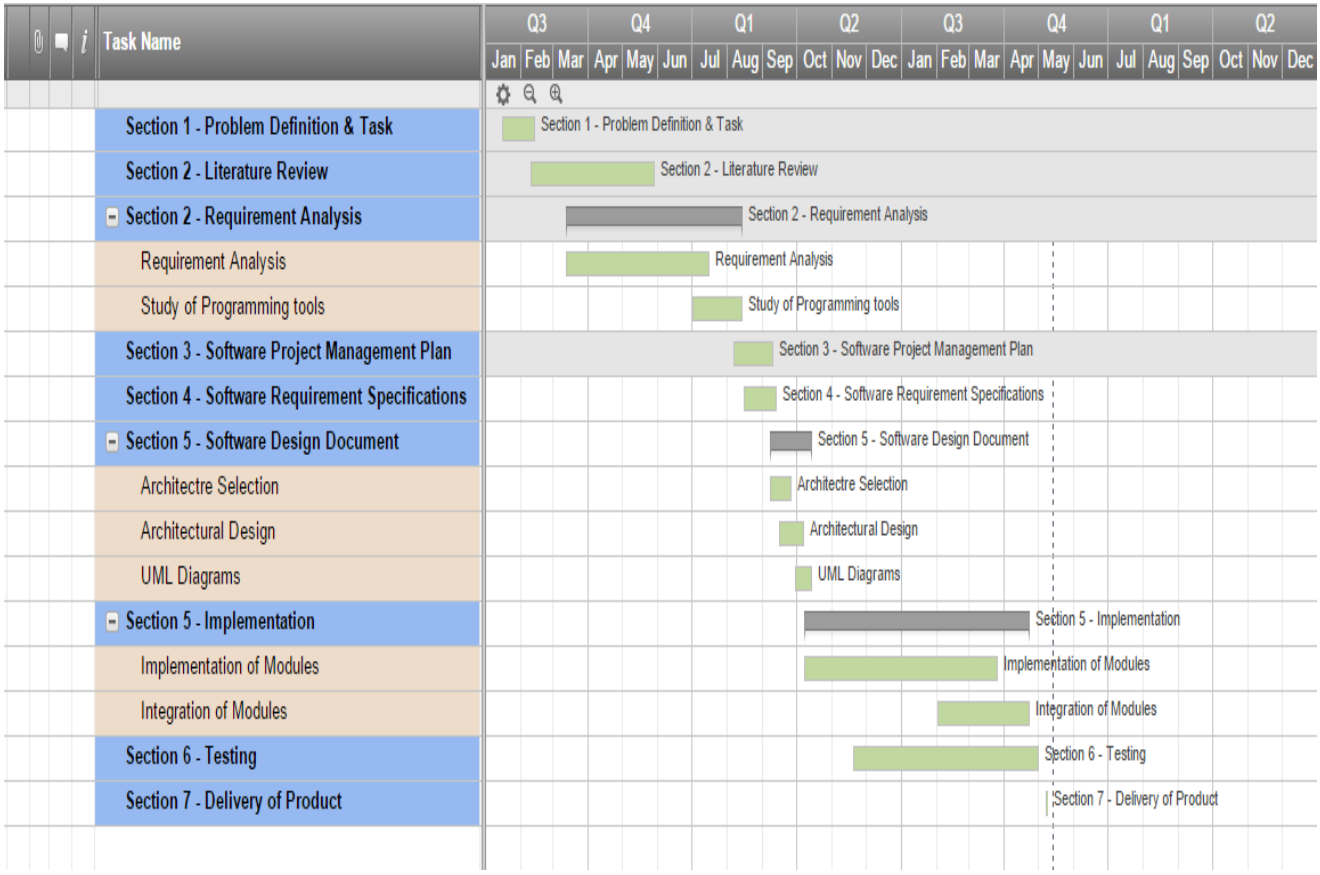


Figure 2 Gantt chart

The above figure shows the predefined time table for the project. The Gantt chart shows different deadlines for different activities. This Gantt chart helps the team to keep track of the progress of the project activities and cope up when the activities are lagging. As seen from the Gantt chart, the expected delivery of the project would be in the month of May [31].

Chapter 5

Project Software Requirement Specifications

5.1. Introduction

The Software Requirement Specification (SRS) is a description of a software system to be developed laying out functional and non-functional requirements.

5.1.1. Product Overview

The Social Media Aggregator is being designed to cater to the needs of the Mumbai Police such that using this application, they can gather information about the various events in the city, people's opinion about Mumbai Police and other such social knowledge from sources such as popular blogs and news websites. The advantage of this application is that the Police personnel need not visit various sites and can view all relevant articles in the same application.

5.2. Specific Requirements

5.2.1. External Interface Requirements

5.2.1.1. User Interfaces

The Social Media Aggregator will be available to the Mumbai Police in the form of a Web Based Application which will contain the following interfaces:

a) A Login Page

This is the first page that appears as soon as the application is launched. User will have to enter his/her mobile number as the userID. For first time user, the admin will provide a temporary password which the user can change overtime. For subsequent visits to the application, mobile number will be used as loginID and user-chosen password will be used as a password.

b) Various Tabs

Upon successful login, the app goes to a new page which shows three main tabs

I. Trending topics

II. Sensitive words

III. Archive

Trending topics tab will show those articles which are most viral on the social Media and are most talked about.

Sensitive words are those special words that are chosen by the programmers to crawl the web pages and generate relevant results. For e.g. Terrorism could be a sensitive word which is used to crawl various web pages and upon choosing this sensitive word, all information related to that word in Mumbai Police's context will appear.

Archive is a special feature of this application where the user can view previously read articles. These articles will be sorted based on the date when they were generated. So the user is required to enter the date to view information of that particular date.

c) Google Cloud Messaging

GCM is a messaging service that would be implemented for the admin of Social Media Aggregator so that he can broadcast important updates to all registered users. The users will have to install GCM application on their smartphones and shall receive the message as a notification.

5.2.1.2. Software Interfaces

Name: MySQL

- Version: 5.6.23

- Purpose: To store the relevant and filtered data by the Crawler and User Information as well.

Name: Eclipse IDE

- Version: 3.5

- Purpose: To develop the Android Application. We also use the Android SDK and AVD(Android Virtual Device). SDK is used to download packages related to the Android development through Eclipse. AVD is used to test the Android Application in a virtual environment.

Name : JCreator

- Version: 5.10

- Purpose: To design the Web Crawler and inject data in MySQL collected by Crawler. This also has a built-in Output console but will not be used.

Name -WAMP (Windows Apache MySQL and Php)

- Version: 2.5

- Purpose: WAMP contains MySQL and PHP which are required for different purposes.

MySQL is used to store the data from the crawler and is used to store the user details which can be accessed later from the dB whenever required. PHP is a server side scripting language and is required for communication with the database and also provides security by abstraction of data. Apache is used to create an environment similar to the client environment and helps in testing. (May not be required later)

5.2.2. Software Product Features

The Application shall provide following features:

1) Validate User

When a user tries to enter the application, his userID and password will be verified and only if the entered details seem valid, access will be granted by navigating to tabs page. If the entered ID and password seem incorrect, three more attempts will be allowed, failing which the app will be locked for a prescribed amount of time.

2) Show Trending Topics Tab

The application shall show a list of trending topics in a separate tab, where all the trending information relevant to Mumbai Police will be available in the form of links. User may click on any of these links to view the entire article.

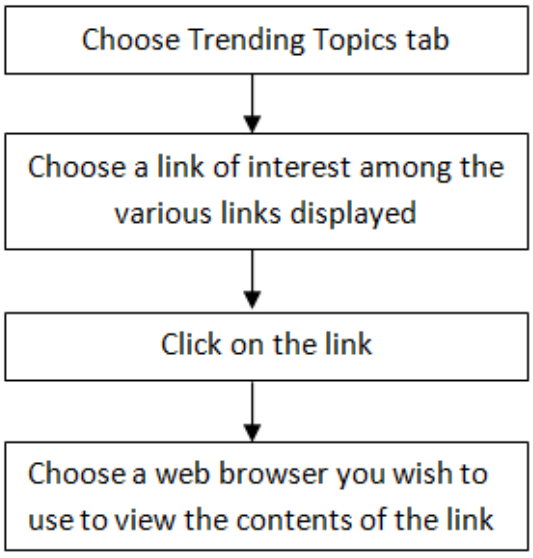


Figure 3: Trending Topics

3) Show Sensitive Words

The App shall show a list of sensitive words prescribed by the Mumbai Police based on which the Crawler will fetch relevant links. The program periodically uses these links to crawl the WebPages and store the links in the database. When a sensitive word is selected, all the links related to that word shall be displayed. User may choose to open any of these links.

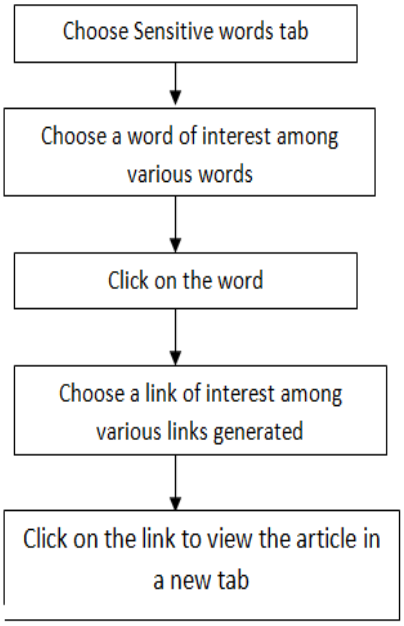


Figure 4: Show Sensitive Words

4) Show Archived Links

This feature shall allow the user to access those links which were generated earlier in time and are no more available in the trending topics or sensitive words list. The user is expected to remember the date when the link was generated, this is because the archived links will be sorted based on the dates of their generation. When the user enters a particular date, all the links generated on that date will be fetched from the database and shown to the user.

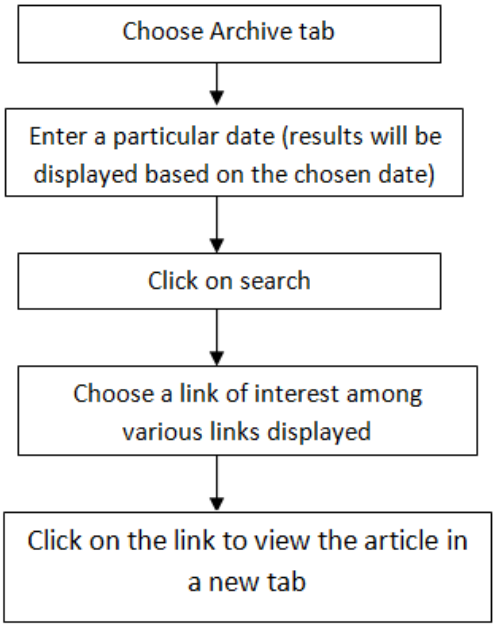


Figure 5: Show Archive links

All the links which will be shown to the user will navigate the user to a new tab. The web browser will then display the contents of each link.

5.2.3. Software System Attributes

5.2.3.1. Reliability

The application is reliable since the crawler will be working all the time and its results will be stored in the database as and when the records are generated. The results generated are however not fully reliable and are not checked for authenticity. Whatever information appears on the social media will be listed and it is upon the Police personnel to decide which information is authentic and which is not.

5.2.3.2. Availability

The application is expected to be available whenever the user is connected to the Internet. The program periodically crawls the prescribed Web Pages and stores the results in a database. The database will be fetched as and when the user requests some information.

5.2.3.3. Security

The security of the application will be ensured since the user ID and password will be first set by an admin which will be given to the user and using these credentials, the user will login for the first time. Thereafter, each user can access the application using his mobile number as the User ID and user-defined password for security. The user ID and password will be stored in a database and password matching will be carried out every time a user enters is password.

5.2.3.4. Portability

The application is designed using Java programming language which is platform independent and hence portable. The program which will crawl the Web pages will run on Java runtime Environment (JRE) and the application for user interaction will be Android OS based and Web based.

5.2.4. Database Requirements

- Data Sets and types: List of sensitive words and generated links. Binary and large object types will be used.
- Availability: Database should be operational 100% of the time means no data loss and little (if any) downtime can be tolerated for proper retrieval of links.
- Actual Disk Space: MYSQL Software will consume 500 MB of the hard disk space and database size may be in the range 1 GB. The indices, support tables, backups, and replication files are usually not considered in actual disk space.
- Backup and Recovery: Backup of the database needs to be taken for proper recovery and it may require 1GB of disk space.
- Growth rate: Links database will increase gradually and this may impact the scalability of the database hardware and software.
- Concurrent users: Only Network admin has rights to access and modify the database.
- Platform requirement: Database will run on all versions of Linux operating system and database version.
- Accessibility: Database can be accessible for at least 2 years.

Chapter 6

Software Design Document

This Software Design Document (SDD) attempts to describe the design specifications used in the development of the Social Media Aggregator. It describes the system and architectural design, as well as the user interface design. It also contains requirement traceability matrix that maps and traces user requirement with test cases.

6.1 Introduction

The purpose of our project is to design a news aggregator for Mumbai Police which crawls through various news websites and blogs, and collects all relevant information to be displayed to the user via a Web UI and an Android UI.

6.1.1 Design Overview

The Social Media Aggregator aims at collecting news articles for police personnel in a way that makes it convenient for them to read articles. To do this, we have divided our project into – modules as follows:

- A Focused Web Crawler
- Admin Panel
- Links Generation and Storing in a Database
- Retrieving Links
- Displaying Links on chosen UI

Table 2 Requirements Traceability Matrix

ID	Requirement Description	Type	Module	Status	Implemented In
R-1	Crawler will visit various pages and store links of those where keywords are found	Functional	Crawler and Storage	In progress	Java
R-2	Every day, UI will display a list of trending topics	Functional	User Interface	Testing	Java
R-3	Only admin has a right to add new sensitive words to the list and allow access to legitimate users.	Non Functional	Admin Panel	Completed	MySQL

6.2 System Architectural Design

6.2.1 Chosen System Architecture

6.2.1.1 System Architectural Style

Following diagram shows communication between different components of the project in C2 Architectural Style focusing on the large-scale of a system.

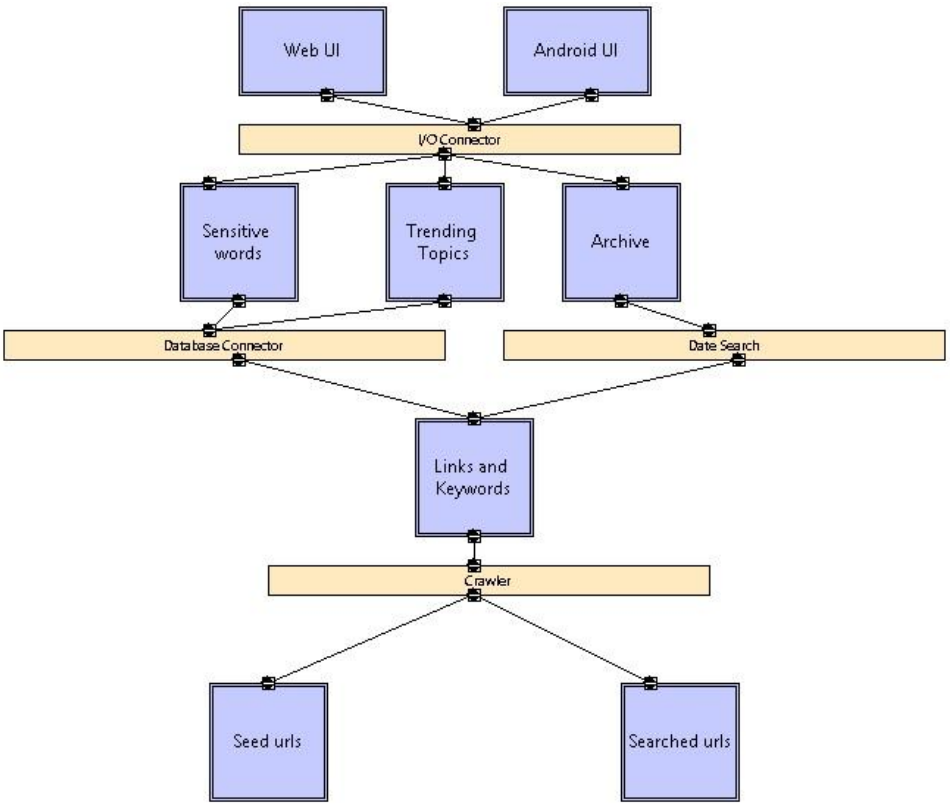


Figure 6 Software Architecture

Components:

1. Web UI and Android UI: interactive screen to the user for input output.
2. Trending Topics: It shows the list of all the trending topics on social sites.
3. Sensitive Words: Shows the list of all keywords predefined by admin.
4. Archive: Stores the links of keywords on the basis of date.
5. Seed URL: Stores the seed URLs.
6. Searched URLs: List of all URLs which the crawler visits.
7. Links and Keywords: List of only the relevant links.

Connectors:

1. I/O: It connects the GUI and the main program. All the results of the main program are sent to the UI using this connector.
2. Crawler: Crawler is used to take the SeedURLs and pass on the generated links for storage
3. Database Connector/search: It helps various modules to connect to the database for storage or searching.

6.2.2 UML Diagrams

A. Use Case Diagram

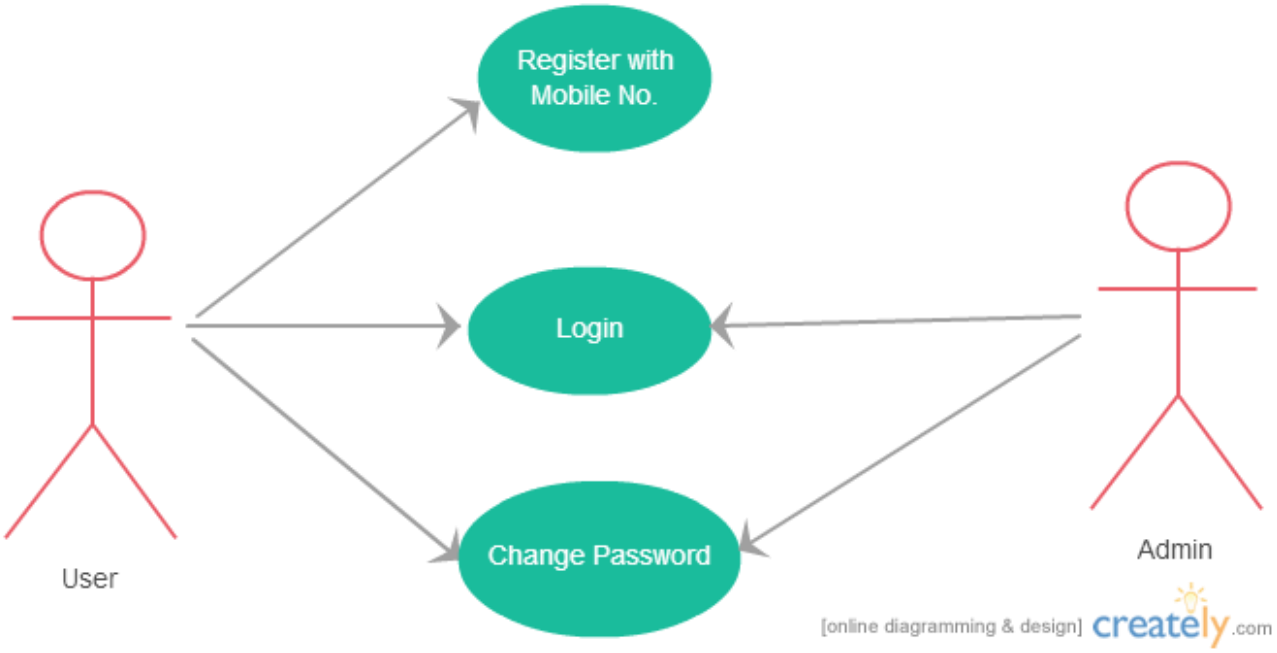


Figure 7 Login Use Case

The use case diagram is used to show the simplest interaction between the admin, user and the system. In this case the various use cases generated are as follows:

- **Register***:* The user registers himself with the application on first time access. Required UserID will be the phone number and password can be any user defined password.
- **Login, Enter Mobile Number, Enter Password:** After registration, the user can login using his login credentials.
- **Change Password:** The user or the admin may change password as and when needed.

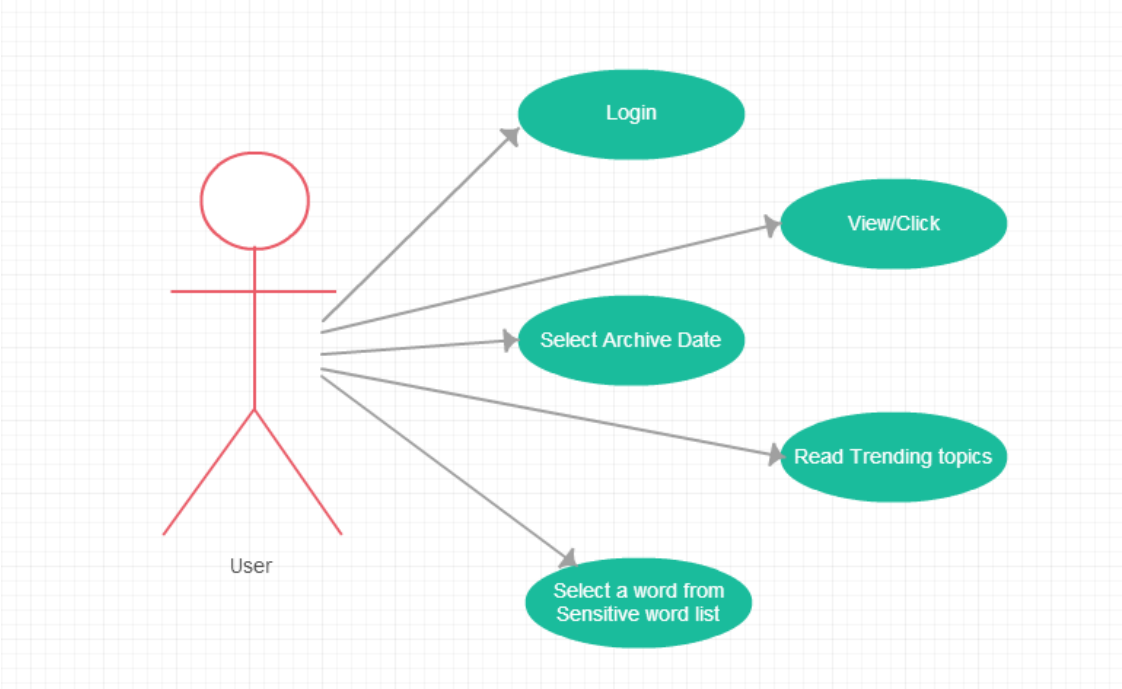


Figure 8 User Use Case

- **Login:** The user first logs into the application by entering correct user name and password.
- **View / Click:** Upon successful login, user is redirected to the tabs page, where he can either view the various results or click on any result to read the contents of the page.
- **Select Archive Date:** If the user chose the archive tab, he needs to enter a particular date from the date picker and upon clicking submit, the results generated on that date will be shown.
- **Read Trending Topics:** The user can choose the trending topics tab and read any of the displayed articles by clicking on the respective link.
- **Select a word from sensitive word list:** The user can select a sensitive word from a list of words such that on clicking, the results based on that word will be displayed.

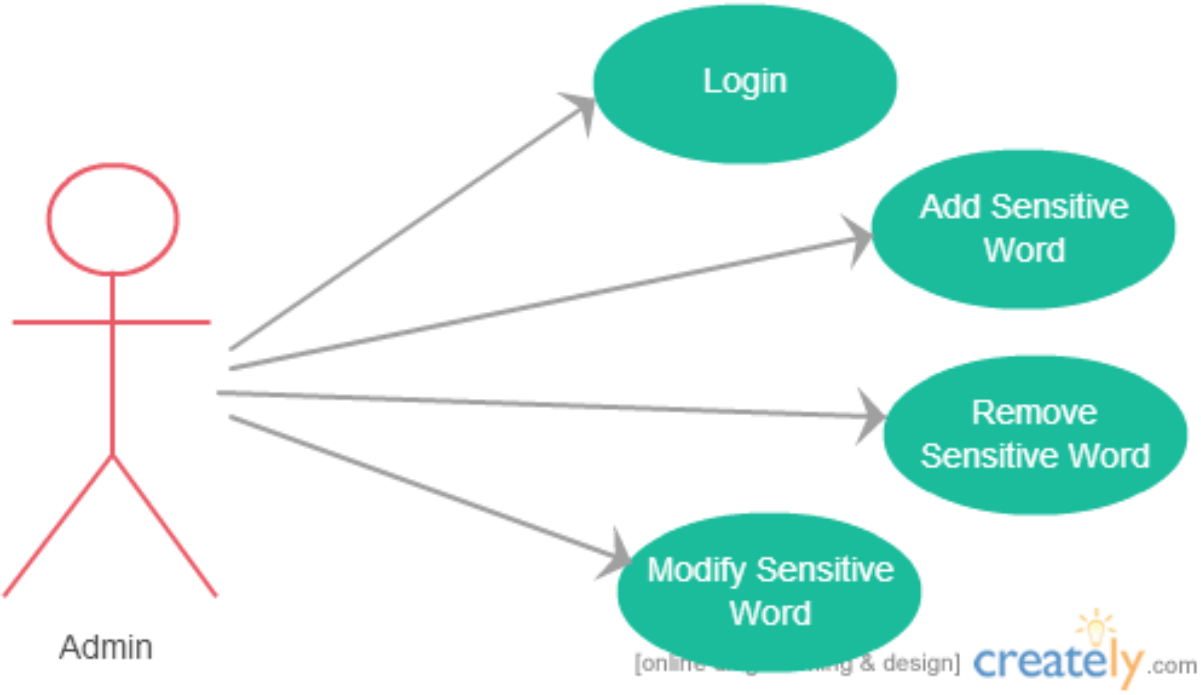


Figure 9 Admin Use Case

- **Login:** The admin can login using his credentials which will allow him to make necessary changes.
- **Add sensitive word:** The admin can add a new sensitive word to the list of words as and when needed.
- **Remove sensitive word:** The admin can remove an existing word from the database as and when needed.
- **Modify sensitive word:** The admin can modify an existing

B. Class Diagram

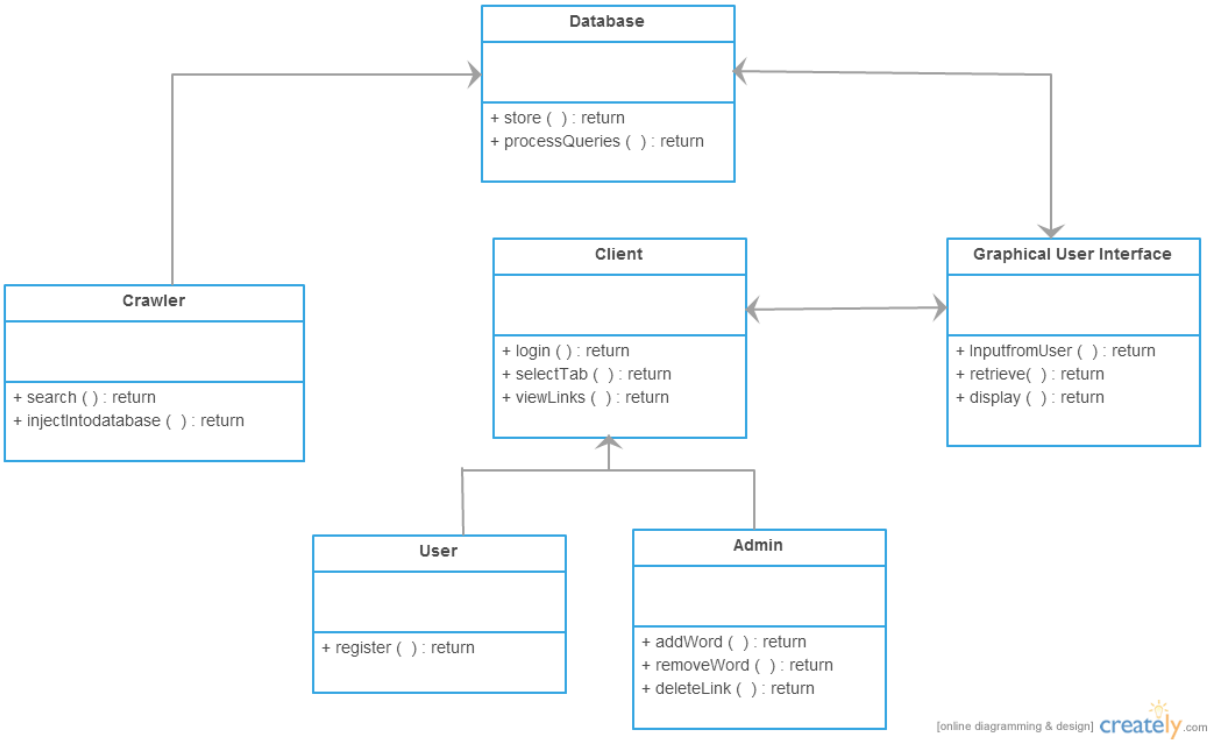


Figure 10 SMA Class Diagram

Following are the classes involved in the system:

- **Crawler:** It is the basic working of the product wherein the crawler searches keyword related articles, feeds the output in the database, retrieves the articles as and when needed.
- **Client:** It is any user who uses the application. The user interacts with the application using the GUI, through which he can login, view various results and logout. The admin on the other hand can do everything that the user does, but moreover, he can add or remove words from sensitive wordlist.
- **Database:** The database is the storehouse of this application. All the output of crawler is stored in the database and the UI takes input from the user and fetches results from the database.
- **GUI:** The GUI allows users to interact with all other components of the application.

C. Component Diagram

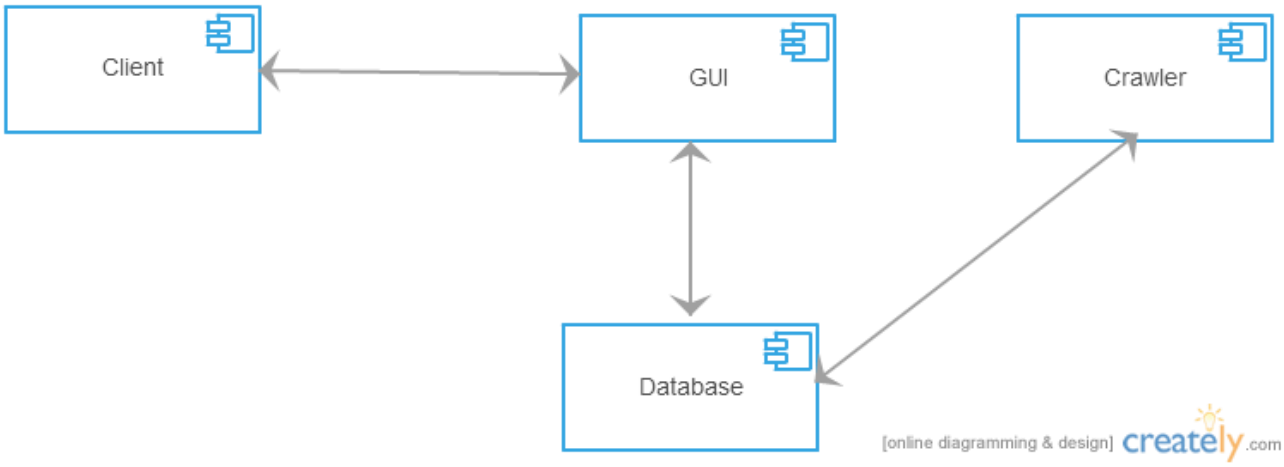


Figure 11 SMA Component Diagram

The various components described here are:

- **Client***:* It is any user who uses the application. The user interacts with the application using the GUI, through which he can login, view various results and logout. The admin on the other hand can do everything that the user does, but moreover, he can add or remove words from sensitive wordlist.
- **Crawler:** It is the basic working of the product wherein the crawler searches keyword related articles, feeds the output in the database, retrieves the articles as and when needed.
- **Database:** The database is the storehouse of this application. All the output of crawler is stored in the database and the UI takes input from the user and fetches results from the database.
- **GUI:** The GUI allows users to interact with all other components of the application.

D. Activity Diagram

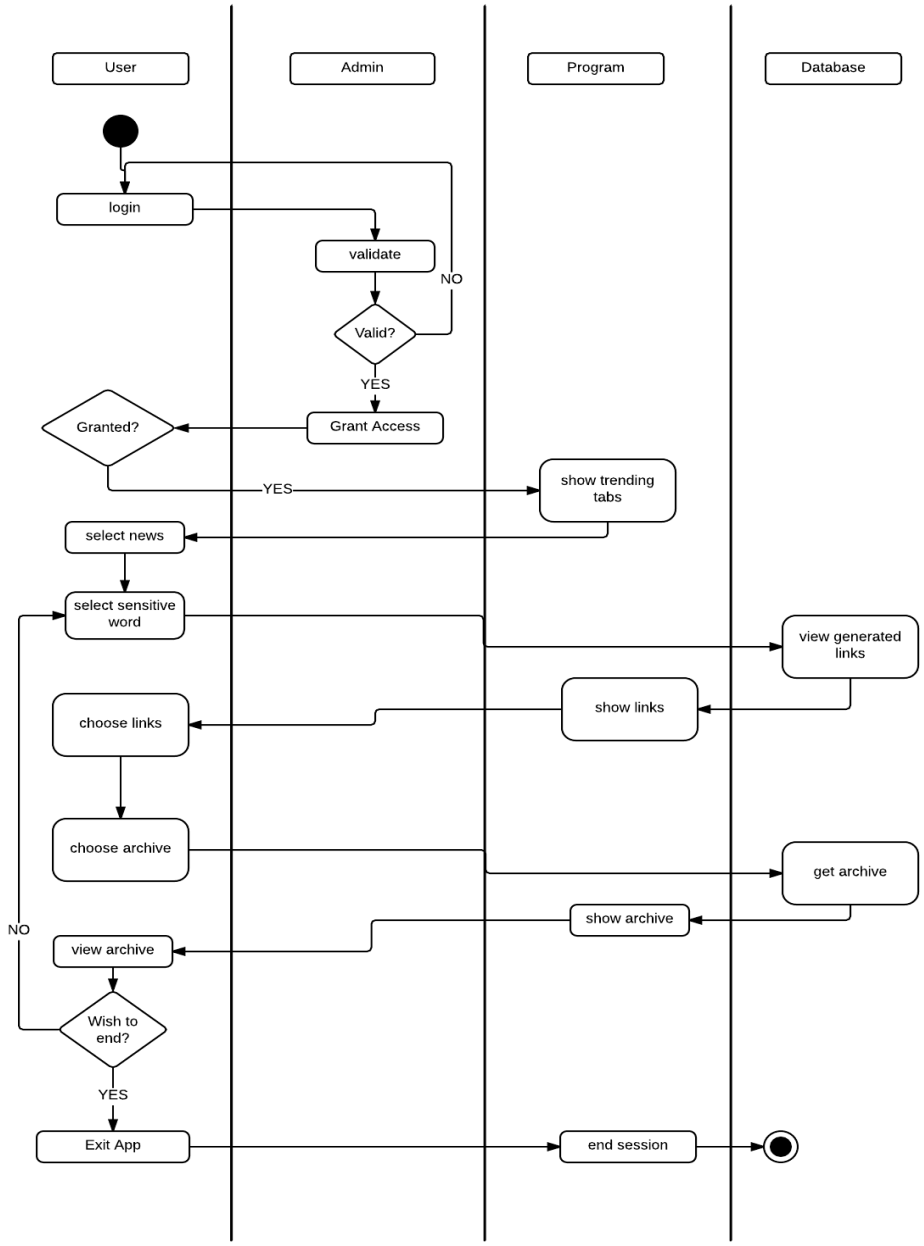


Figure 12 User Activity Diagram

The activity diagram shows the various activities that take place when a user interacts with the application.

E. Sequence Diagram

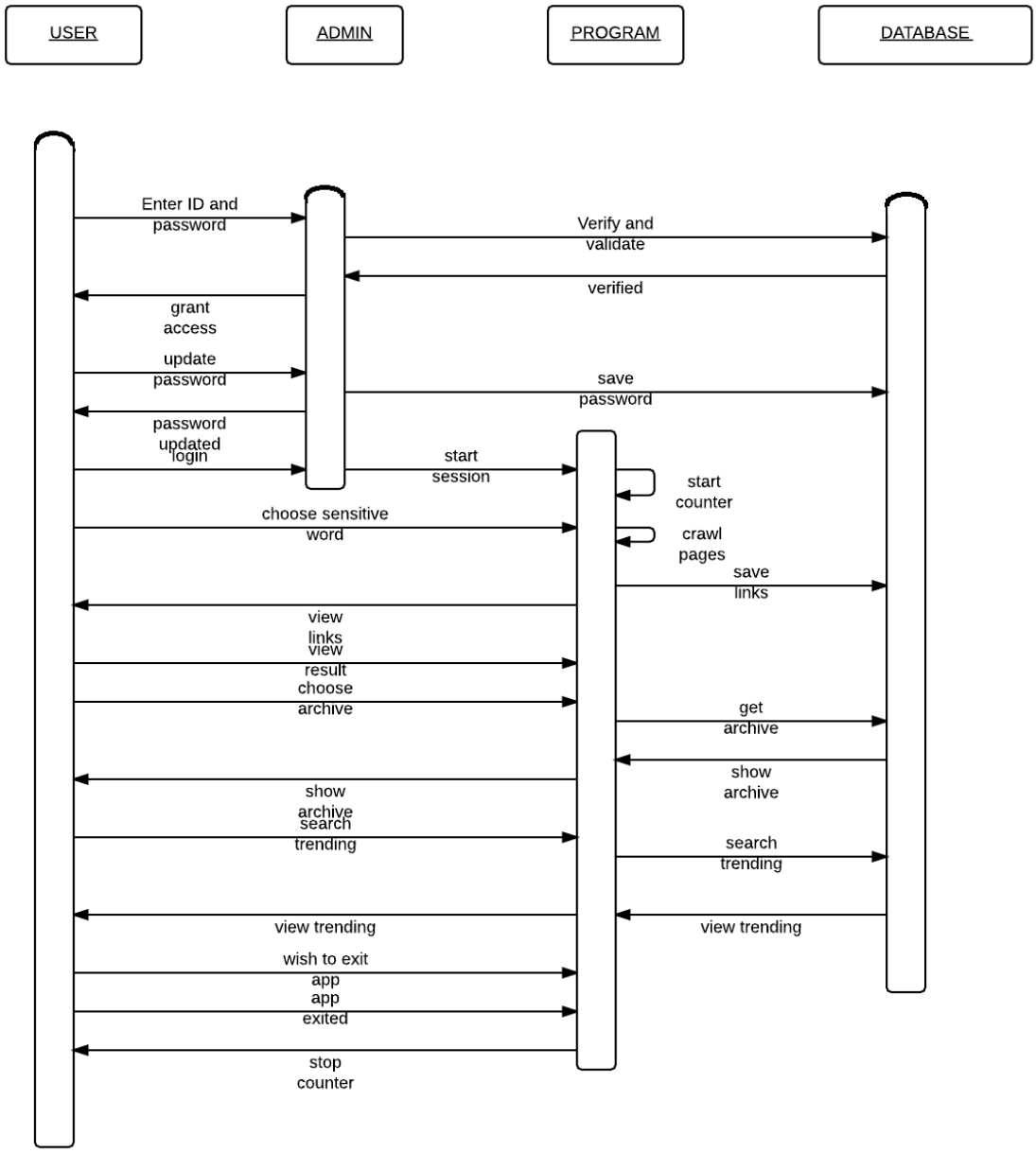


Figure 13 SMA Sequence Diagram

The sequence diagram shows the sequence of events which take place in a session and shows the involvement of various entities.

6.2.3 Design of Crawler

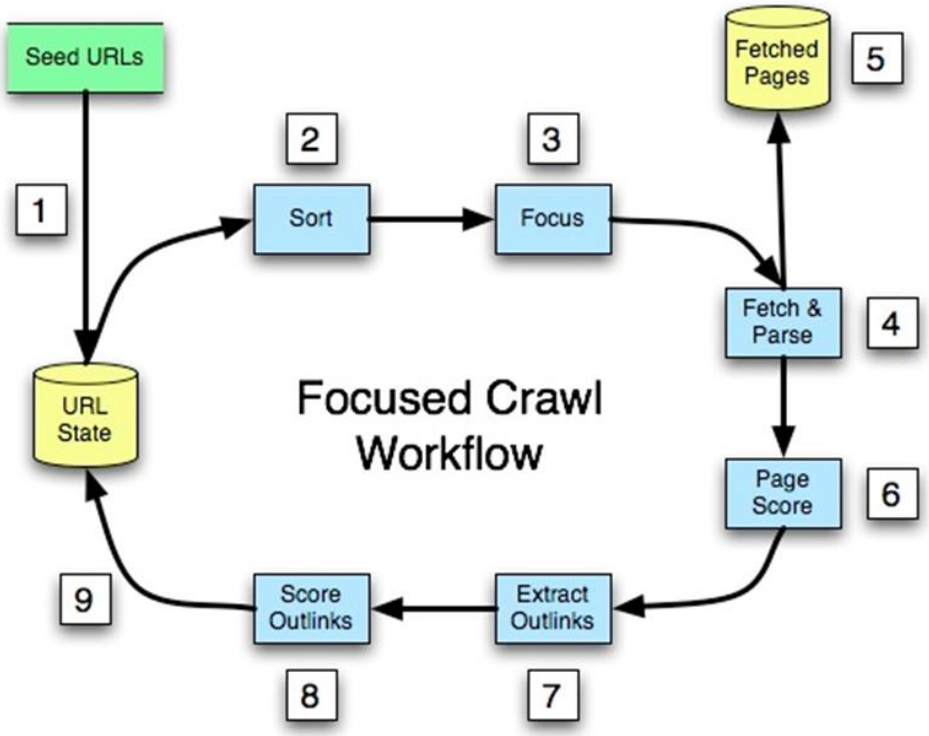


Figure 14 Architecture of Web Crawler [34]

Initially, the URL State database is loaded with a set of URLs. These URLs can be a broad set of domains with the highest traffic or the result from some selective searches against some other index or manually handpicked URLs that point to the specific high quality pages. Once the URL state database is loaded, the first loop of the crawler begins. The prime step of all the loops is to extract all the unprocessed URLs and sort them according to their frequency score. Next is the critical step of deciding which how many URLs to process further in this loop. The fewer the URLs, the tighter the focus of the crawler. The selection can be based on maximum frequency of the keyword on the page.

Now, having the set of accepted URLs the fetching process begins which entails all of the usual steps required for polite & efficient fetching. Pages that are fetched are normally stored in the fetched pages database and are then parsed. The content of the parsed page is given to the frequency counter, which returns a value representing the frequency of occurrence of the keyword on the page. Finally,

the URL State database is updated with the results of fetch attempts (succeeded, failed), all newly discovered URLs are added. At this point the focused crawler can terminate, if sufficient pages of high enough quality have been found, or the next loop can begin. In this manner the crawl proceeds in a breadth-first manner, focusing on areas of the web graph where the most high frequency scoring pages are found.

6.2.4 User Interfaces

The Social Media Aggregator will be available to the Mumbai Police in the form of a Web Based Application which will contain the following interfaces:

a) A Login Page

This is the first page that appears as soon as the application is launched. User will have to enter his/her mobile number as the userID. For first time user, the admin will provide a temporary password which the user can change overtime. For subsequent visits to the application, mobile number will be used as loginID and user-chosen password will be used as a password.

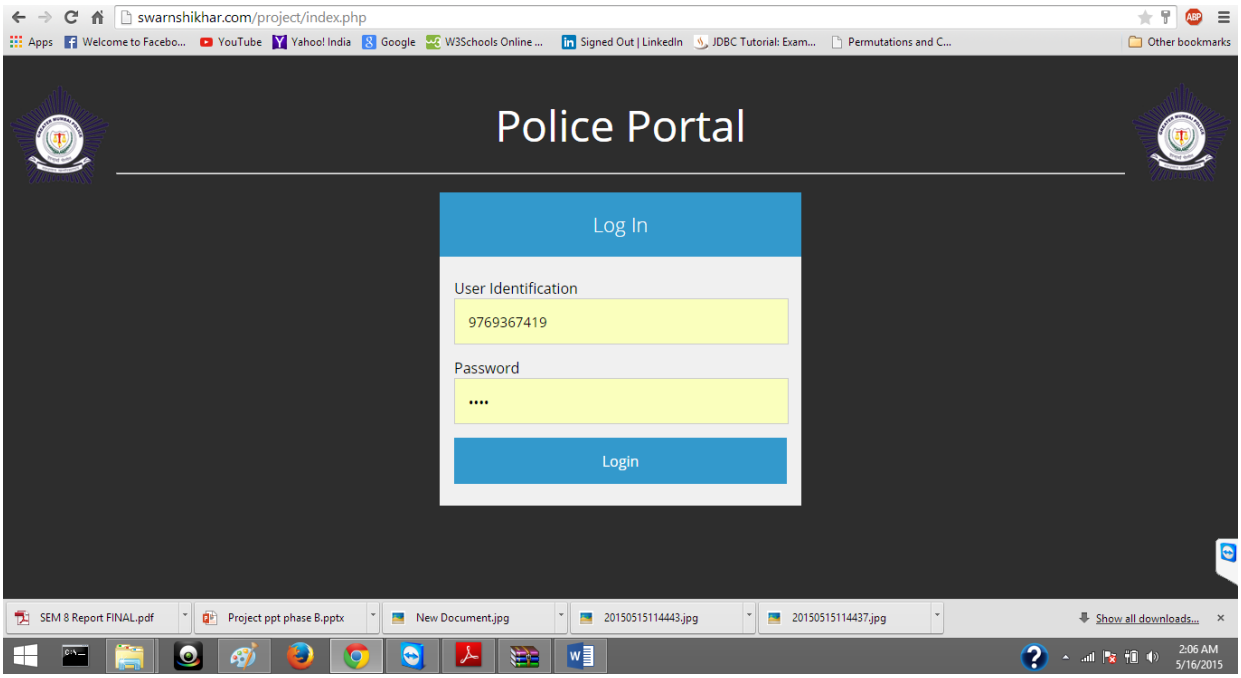


Figure 15 Login Page of SMA Web Portal

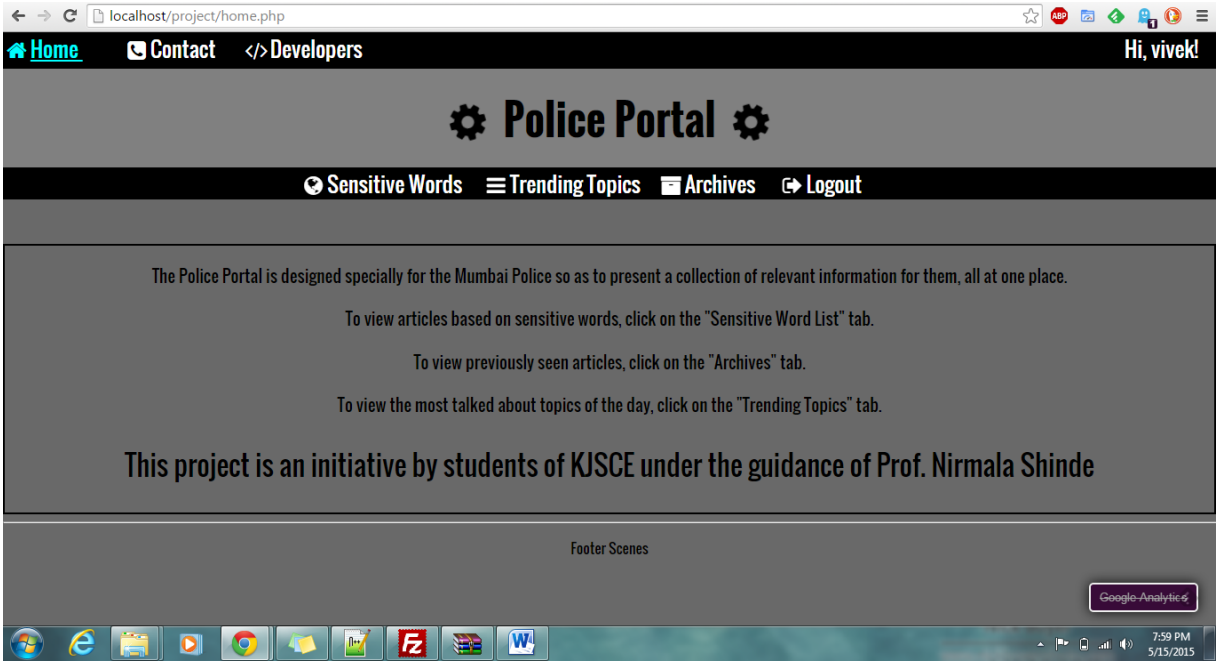


Figure 16 Home Page of SMA Web Portal

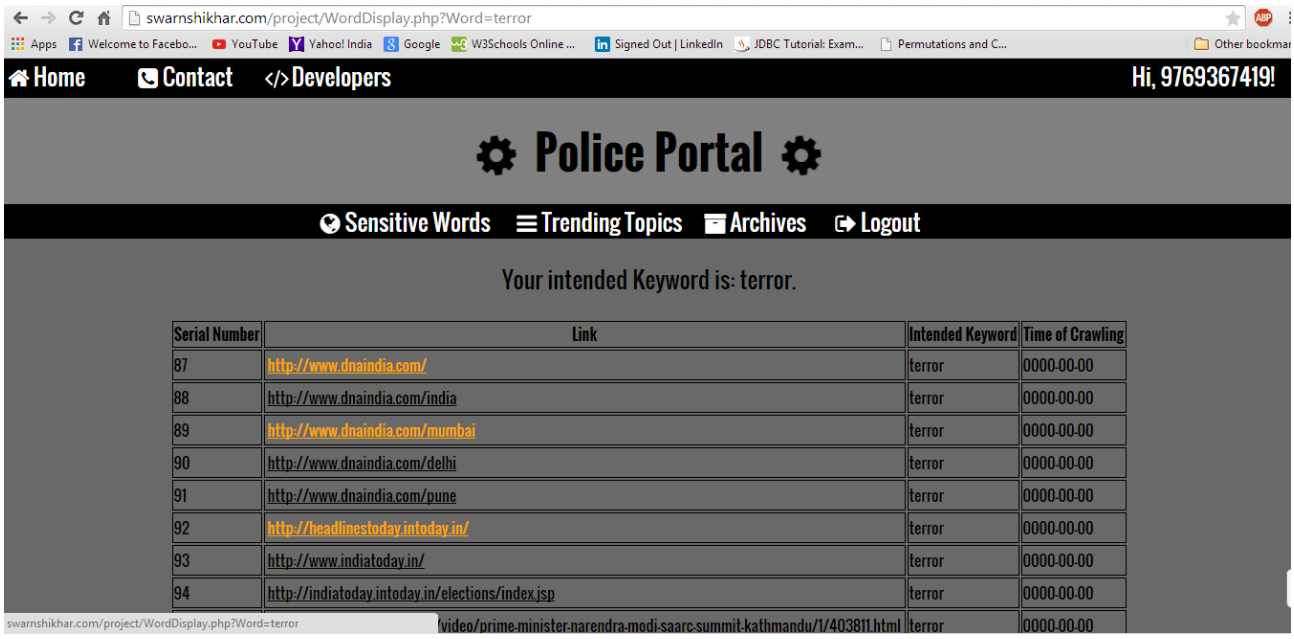


Figure 17 Retrieved Links from the Database

b) Various Tabs

Upon successful login, the app goes to a new page which shows three main tabs

I. Trending topics

Trending topics tab will show those articles which are most viral on the social Media and are most talked about. On clicking on Trending Topics tab the following Page appears :

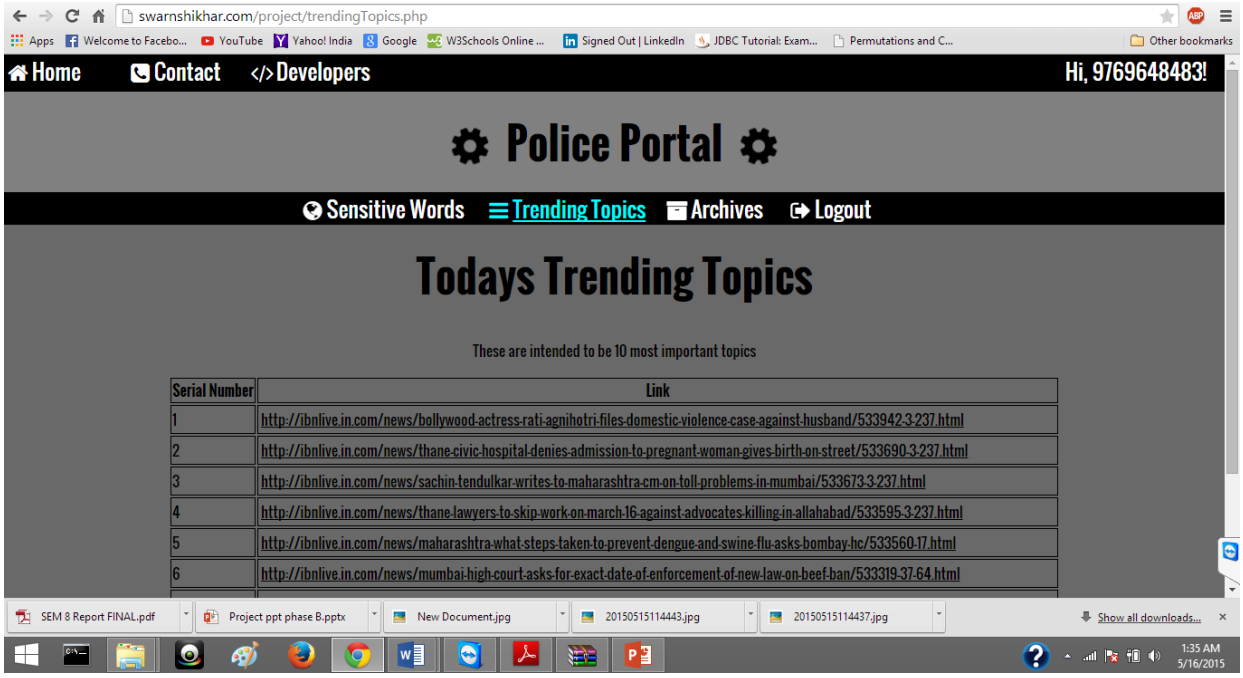


Figure 18 Trending Topics Page of SMA Web portal

II. Sensitive words

Sensitive words are those special words that are chosen by the programmers to crawl the web pages and generate relevant results. For e.g. Terrorism could be a sensitive word which is used to crawl various webpages and upon choosing this sensitive word, all information related to that word in Mumbai Police's context will appear. On clicking on Sensitive words tab the following Page appears and allows the user to choose the sensitive word from the list.

Figure 19 Sensitive Words Page for SMA Web Portal

III. Archive

Archive is a special feature of this application where the user can view previously read articles. These articles will be sorted based on the date when they were generated. So the user is required to enter the date to view information of that particular date. On clicking on Archive tab and selecting a date the following page appears.

C) Google Cloud Messaging:

GCM is a messaging service implemented for the admin of Social Media Aggregator so that he can broadcast important updates to all registered users. The users will install GCM application on their smartphones and will receive the message as a notification

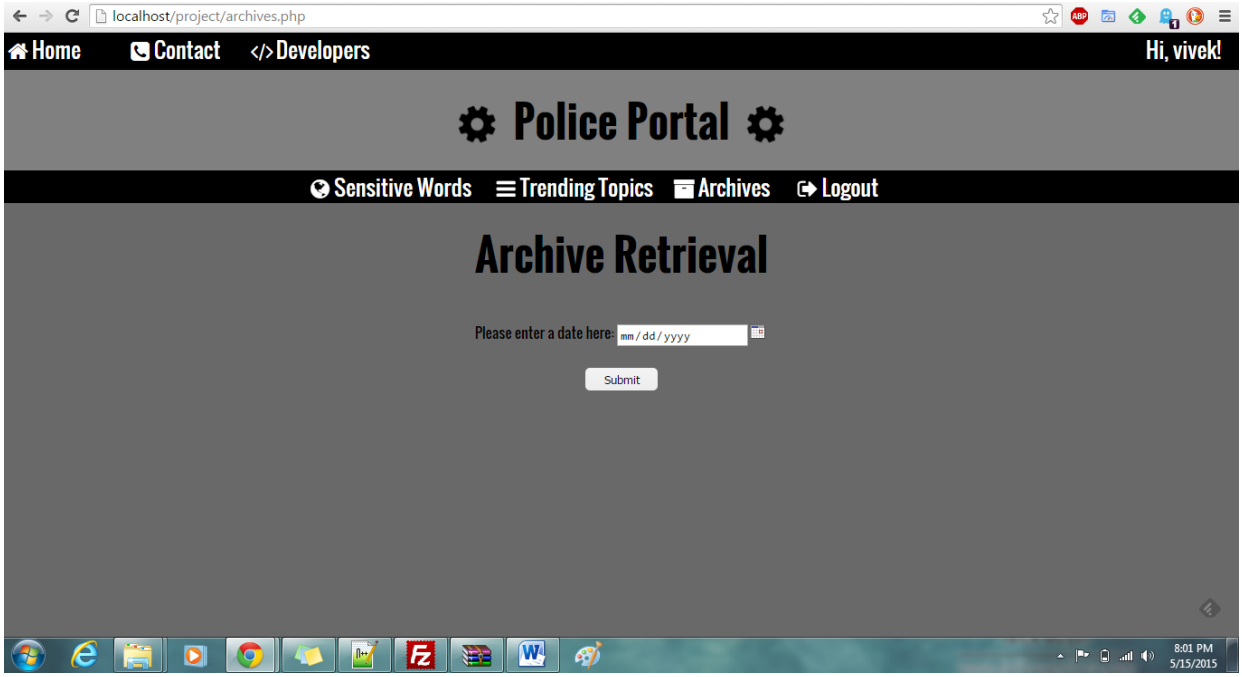
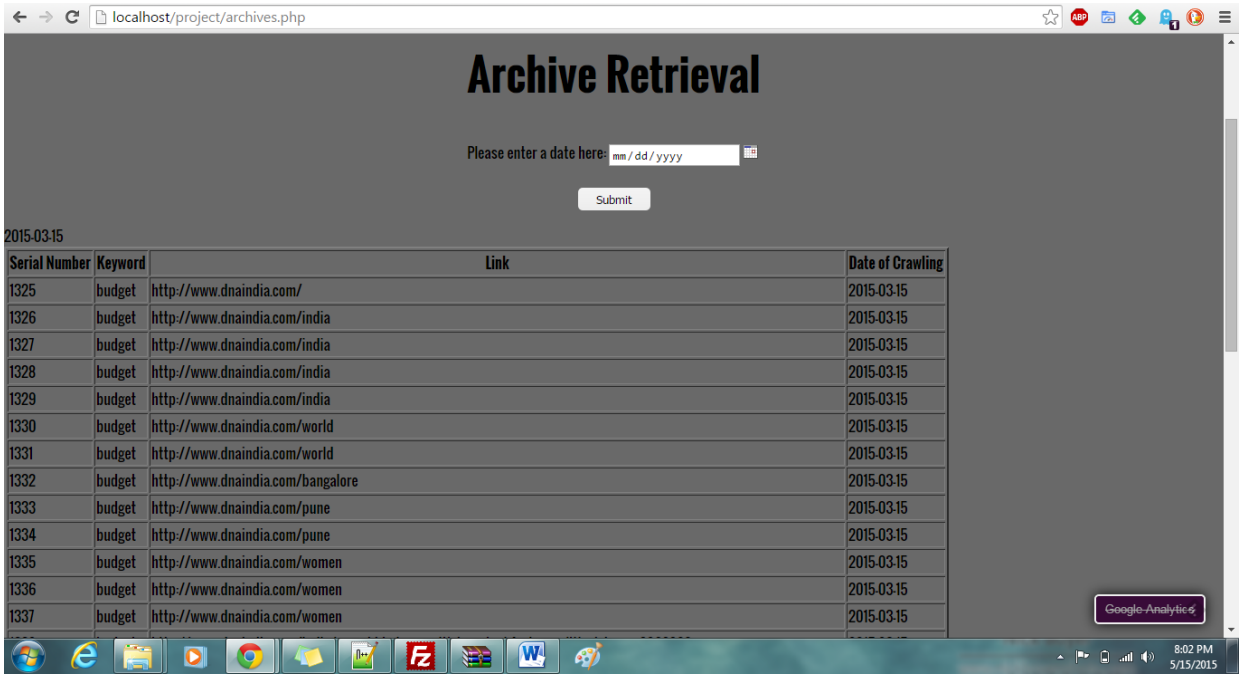


Figure 20 Archive Page for SMA Web Portal 1



Serial Number	Keyword	Link	Date of Crawling
1325	budget	http://www.dnaindia.com/	2015-03-15
1326	budget	http://www.dnaindia.com/india	2015-03-15
1327	budget	http://www.dnaindia.com/india	2015-03-15
1328	budget	http://www.dnaindia.com/india	2015-03-15
1329	budget	http://www.dnaindia.com/india	2015-03-15
1330	budget	http://www.dnaindia.com/world	2015-03-15
1331	budget	http://www.dnaindia.com/world	2015-03-15
1332	budget	http://www.dnaindia.com/bangalore	2015-03-15
1333	budget	http://www.dnaindia.com/pune	2015-03-15
1334	budget	http://www.dnaindia.com/pune	2015-03-15
1335	budget	http://www.dnaindia.com/women	2015-03-15
1336	budget	http://www.dnaindia.com/women	2015-03-15
1337	budget	http://www.dnaindia.com/women	2015-03-15

Figure 21 Archive Page for SMA Web Portal 2

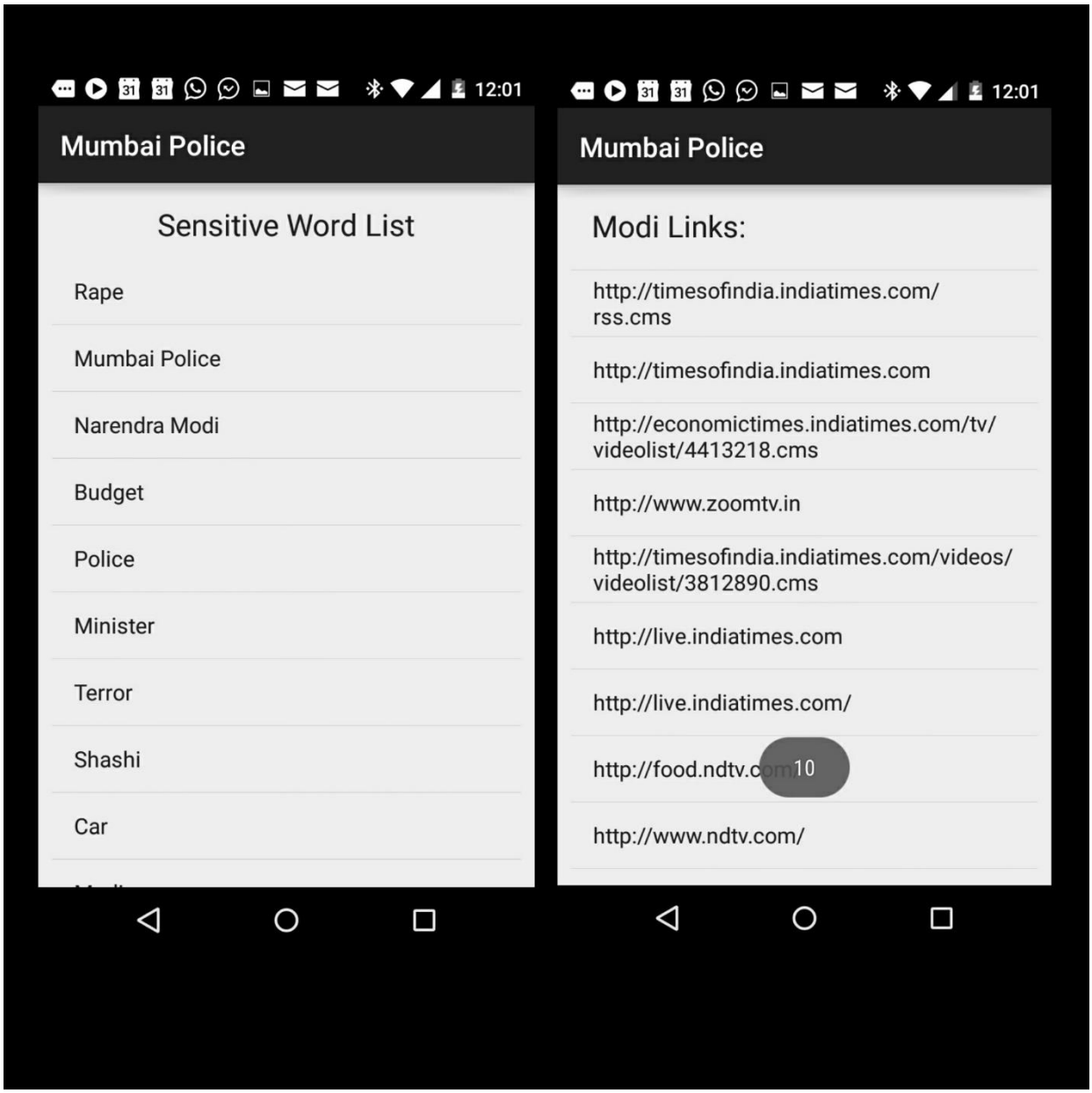


Figure 22 Sensitive Words on Android UI

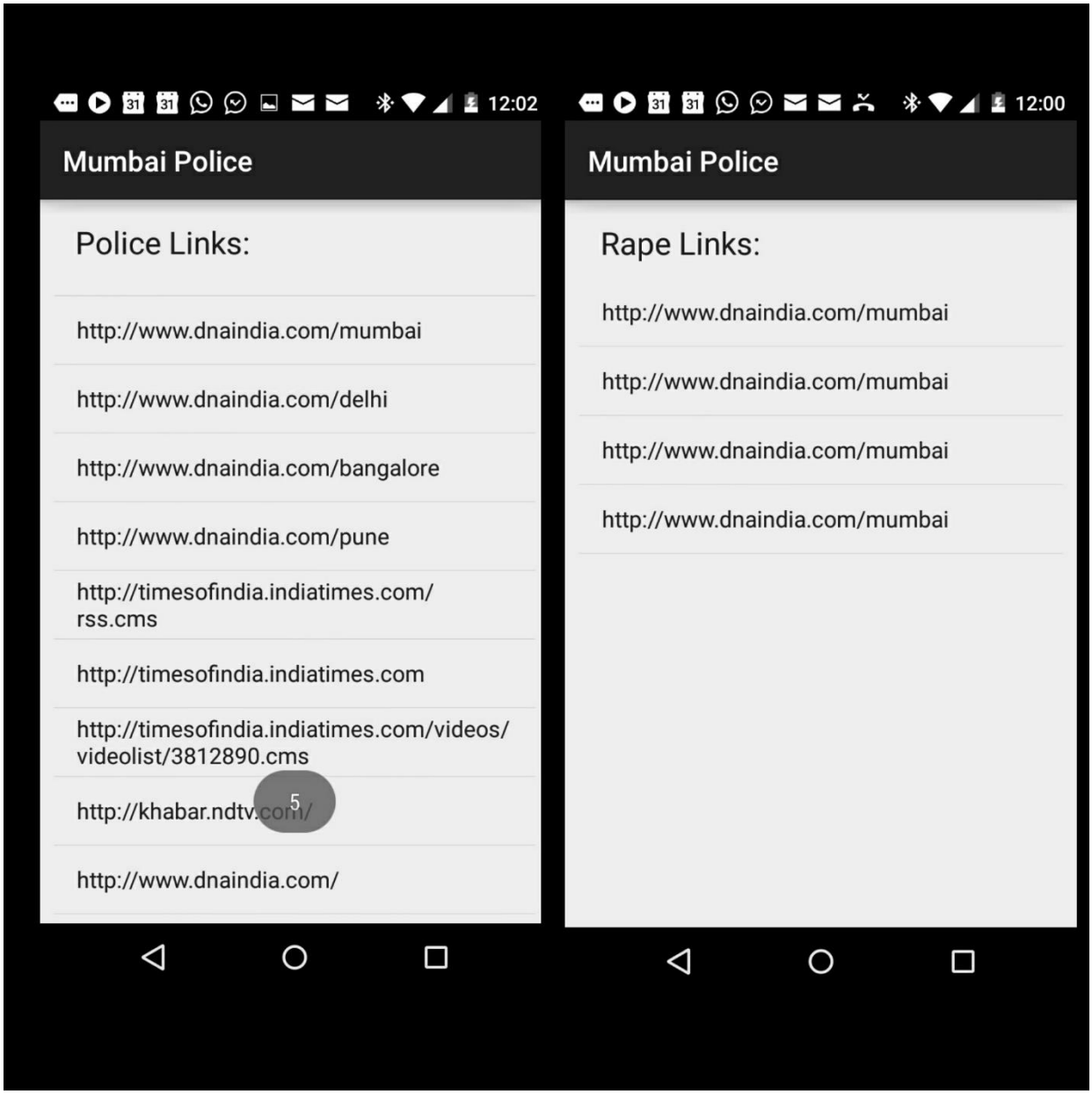


Figure 23 Sensitive Words on Android UI

Chapter 7

Software Testing Document

7.1 Testing Approaches

White box testing:

Using white-box testing methods, the software engineer can derive test cases that do all of the following:

1. Exercise all independent paths within the module at least once.
2. Exercise all logical decisions for both true and false scenarios.
3. Execute all loops at their boundaries and within their operational loops
4. Exercise all internal data structures to ensure their validity.

Black box testing:

Black box testing attempts to find errors in the following categories:

1. Incorrect or missing functions
2. Interface errors
3. Errors in data structures or external database access
4. Performance errors
5. Initialization and termination errors

Unit testing:

Unit testing focuses verification effort on the smallest unit of software design – the software module. Using the component level design description as a guide, important control paths are tested to uncover errors within the boundary of the module. The unit test is white box oriented, and the steps can be conducted in parallel for multiple modules.

Integration testing:

Interfacing of various modules can cause problems. Data can be lost across an interface, one module may affect the other, and individually acceptable imprecision may be magnified when combined. Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. The objective is to take unit tested components and build a program structure that has been dictated by design.

Stress testing:

During earlier testing steps, white box and black box techniques result in a thorough evaluation of normal program functions and performance. Stress tests are designed to confront programs with abnormal situations. Stress testing executes a system in a manner that demands resources in abnormal quantity, frequency or volume. Essentially, the tester attempts to break the program.

Performance testing:

Software that performs the required functions but does not conform to performance requirements is unacceptable. Performance testing is designed to test run-time performance of software within the context of an integrated system. Performance testing occurs through all the steps in the testing process. However, it is not until all system elements are fully integrated that the true performance of a system can be ascertained.

Security testing:

Any computer-based system managing sensitive information or causes actions that can harm individuals is a target for improper penetration. Security testing attempts to verify that protection mechanisms built into a system will, in fact, protect it from improper penetration. During security testing, the tester plays the role of the hacker who desires to penetrate the system. Given enough time and resources, good security testing will ultimately penetrate a system. The role of the system designer is to make penetration cost more than the value of the information that will be obtained.

Recovery testing:

Many computer-based systems must recover from faults and resume processing within a pre-specified time. In some cases, a system must be fault-tolerant, i.e. processing faults must not cause overall system function to cease. In other cases, a system failure must be corrected within a specified period of time or severe economic damage will occur. Recovery testing is a system test that forces the software to fail in a variety of ways and verifies that recovery is properly performed. If recovery is automatic, re-initialization, check-pointing mechanisms, data recovery and restart are evaluated for correctness. If recovery requires human intervention, the mean-time-to-repair (MTTR) is evaluated to determine whether it is within acceptable limits.

7.2 Testing Plan

It is a document consisting of different test cases designed for different testing objects and different testing attributes. The plan puts the test in sequential order as per the strategies chosen that is, top down and bottom up. The test plan is matrix of test cases listed in order of its execution.

Test Plan is developed to detect and identify potential problems before delivering the Software to its users. The scope of test will be limited just to the boundaries as the white box testing cannot be in detail.

Table 3 Testing Plan

Type of Testing	Feature to be tested	Responsibility allocation	Testing tools and environment	Time required
Unit testing	Crawler execution	Rhythm Shah	JCreator	5 Days
	DB Management		SQL Server	
Integration Testing	1.Functions of the various modules acting together	Riya Patni	Web Browser	9 Days
	2. Interface to Application		Wamp Server	
	3. Links between Application		Wamp Server	
System Testing	1. Relevance of the results generated	Vivek Patani	Wamp Server	6 Days
	2. Recovery in the event of failure	Vruksha Shah		
	3. Security in the system from illegal access.	Rhythm Shah		

Table 3 Testing Plan (Continued)

Regression Testing	1. Resolve problems due to error	Vivek Patani	Wamp Server	3 Days
	2. Ensure no new errors are Generated			
Alpha testing	1.Authentication	Vruksha Shah	Manual	5 Days
	2. Critical process and functions		Wamp Server	
Beta testing	Monitored overall System	Rhythm Shah	Manual	3days

7.3 Test Cases

A test case, in software engineering, is a set of conditions or variables under which a tester will determine whether an application, software system or one of its features is working as it was originally established for it to do. The mechanism for determining whether a software program or system has passed or failed such a test is known as a test oracle. In some settings, an oracle could be a requirement or a use case, while in others it could be a heuristic. It may take many test cases to determine that a software program or system is considered sufficiently scrutinized to be released. Test cases are often referred to as test scripts, particularly when written - when they are usually collected into test suites.

Table 4 Test Cases

Type of Testing	Items to be tested	Input	Expected Output	Actual output
Unit testing	1.User Interface	Query	URLs	URLs
		Query1	Not found	URLs
		Query2	URLs	URLs
	1.Connectivity	Click on Hyperlink	New page Opens	New page opens

Table 4 Test Cases (Continued)

Integration Testing	2. Querying	Post query	Triggers an event	Related query gets executed
		Post query	Data not found	Data not found
System Testing	1. Login failure		Backup system at work	Backup system at work
Alpha testing	1.Authentication	Password Entry	Password accepted safely	Password accepted safely
		Password Entry	Wrong password	Login rejected

Chapter 8

Conclusion and Future Work

Social Media Aggregator acts as a news aggregator for Mumbai Police wherein important news articles based on predefined list of sensitive words are shown to the user. These articles are searched using a focused crawler which uses three seed links to start crawling and thereafter generates keyword related results up to a certain limit. The focused crawler uses breadth first search algorithm and goes up to level 2 where seed links are at level 0.

The results of the crawler are displayed to the user in three ways: sensitive word based results, trending topics, archive. There is a separate admin panel which the admin can use to modify the list of sensitive words.
Google Cloud Messaging (GCM) module is also implemented which can be used by the admin to send important updates to users in the form of message pop ups. The application reduces the police personnel's efforts by combining several results and displaying everything together.

Future Work

Future work includes improving the crawler efficiency to speed up the crawling process. A major scope for future work is to have extensive tests with a large volume of web pages and include natural language programming semantics such that quality of the results is further improved by generating only those results which are semantically relevant.

Chapter 9

References

1. Mehdi Naghavi and Mohsen Sharifi, “A Proposed Architecture For Continuous Web Monitoring Through Online Crawling Of Blogs”, 2013.
2. Jain Nidhi, Rawat Paramjeet,” A Study of Focused Web Crawlers for Semantic Web”, International Journal of Computer Science and Information Technologies, Vol. 4 (3) , 2013, 398-402.
3. Alexander Shen “Algorithms and Programming: Problems and solutions” Second edition Springer 2012, pg 135
4. Narasingh Deo “Graph theory with applications to engineering and computer science” PHI, 2004 Pg 301
5. Ben Coppin “Artificial Intelligence illuminated” Jones and Barlett Publishers, 2004, Pg 77.
6. TIAN Chong “A Kind of Algorithm For Page Ranking Based on Classified Tree In Search Engine” Proc International Conference on Computer Application and System Modeling (ICCASM 2010)
7. Andy Yoo,Edmond Chow, Keith Henderson, William McLendon,Bruce Hendrickson, ÄUmit CatalyÄurek “A Scalable Distributed Parallel Breadth-First Search Algorithm on BlueGene/L” ACM 2005.
8. J.Kleinberg “Authoritative sources in a hyperlinked environment”, Proc 9th ACM-SIAM Symposium on Discrete Algorithms, 1998.
9. Shaojie Qiao, Tianrui Li, Hong Li and Yan Zhu, Jing Peng, Jiangtao Qiu “SimRank: A Page Rank Approach based on similarity measure” 2010 IEEE
10. S. N. Sivanandam, S. N. Deepa “Introduction to Genetic Algorithms” Springer, 2008, pg 20

11. Harry Zhang “The Optimality of Naive Bayes” American Association for Artificial Intelligence 2004.
12. S.N. Palod, Dr S.K.Shrivastav,Dr P.K.Purohit “Review of Genetic Algorithm based face recognition” International Journal of Engineering Science and Technology (IJEST) Vol. 3 No. 2 Feb 2011
13. Banu Wirawan Yohanes, Handoko, Hartanto Kusuma Wardana,” Focused Crawler Optimization Using Genetic Algorithm”, 2011.
14. Apoorv Vikram Singh,Vikas,Achyut Mishra, “ A Review of Web Crawler Algorithms” ,International Journal of Computer Science and Information Technologies, Vol. 5 (5) , 2014, 6689-6691
15. Swati Mali, B.B.Meshram, "Focused Web Crawler with Page Change Detection Policy" in 2nd International Conference and workshop on Emerging Trends in Technology (ICWET) Proceedings published by International Journal of Computer Applications® (IJCA), 2011.
16. S. Chakrabarti, M. Van Den Berg, B. Dom, “Focused Crawling: A New Approach to Topic specific web resource discovery”, Proc. Of 8th International WWW conference, Toronto, Canada, May,1999.
17. Soltiris Batsakis, Euripides G>M Petrakis, Evangelos Milos, “Improving the Performance of Focused Crawlers”.
18. Gupta G.K, “Introduction to Data Mining with Case Studies”, Third Edition,2013.
19. Niharika Sachdeva, Ponnurangam Kumaraguru. “Online Social Media and Police in India: Behaviour, Perceptions, Challenges”,2014.
20. Nikolaos P.,Georgios k., Efstathios S, “An Agent – Based Focussed Crawlng Frammework For Topic- and Genre- Related Web Document”,2012.
21. Michael Burton , Donn Felker, “Android Application Development For Dummies”, 2013.
22. Weicheng Ma,Xiuxia Chen, WEnquian Shang,” Advanced deep web crawler based on Dom”,2012.
23. Thomas A. Powell, “Web Design : The Complete Reference” , Third Edition,2000.

24. S.N.Sivanandanam , S.N. Deepa “principles of Soft Computing”, Second Edition.
25. http://en.wikibooks.org/wiki/Introduction_to_Software_Engineering/Process/Methodology (last accessed on 25th September 2015)
26. http://www.w3schools.com/ (last accessed on 13th May 2015)
27. http://www.uml-diagrams.org/ (last accessed on 5th September 2015)
28. https://developer.android.com/training/index.html (last accessed on 13th May 2015)
29. http://www.androidhive.info/ (last accessed on (last accessed on 13th May 2015)
30. https://creately.com/app/# (last accessed on 5th September 2015)
31. https://www.smartsheet.com/ (last accessed on 13th May 2015)
32. http://ssltest.cs.umd.edu/~sayyadi/files/papers/4A_Method_for_Focused_Crawling Using_Combination_of_Link_Structure_and_Content_Similarity.pdf(last accessed on 13th May 2015)
33. http://www.researchgate.net/publication/261347885_Efficient_focused_crawling_based_on_best_first_search(last accessed on 13th May 2015)
34. http://www.scaleunlimited.com/about/focused-crawler/

APPENDIX

I. Minimum System Requirement:

- Windows 7 Operating System with Intel Pentium 2.2 GHz CPU and 1GB of RAM.
- System should have JDK 6 and above.
- WAMP server should be installed on the system.
- MySQL should be installed on the system.
- Computer should have internet connection with 512kbps (minimum) speed.

II. User's Manual:

1. Now open the browser and type "www.swarnashi.com/project1" in address bar and hit enter key.
2. You will be directed to home page, where you will be required to register to the system.
3. After successful completion of registration process you will be redirected to the main page where you can choose required feature to obtain desired results.

III. Technical Reference Manual:

Install JDK (6 or above):

1. Download the JDK from http://www.oracle.com/technetwork/java/javase/downloads/index.html , website by default will give the latest version which is surely above 6.0 and save that file (.exe) on your computer.
2. Double click on that file and install it.
3. After successful installation it is ready for use.

Install WampServer:

1. Download WampServer from official WampServer Page
   http://www.wampserver.com/en/
2. Double click on the downloaded file and install it.
3. After successful installation WampServer is ready to use for creating and managing databases.

Setting up an Android Development Environment:

1. Download an IDE
2. Install Eclipse.
3. Download and install the Java Runtime Environment(JRE).
4. Download and install Java Development Kit(JDK).
5. Install Android SDK.
6. Download and Install the Android Development Tools (ADT) plugin for Eclipse.
7. The setup is now ready to use.

Papers Published

- Vruksha Shah, Riya Patni, Vivek Patani, Rhythm Shah,"**Understanding the focused Crawler**", International Journal of Computer Science and Information Technologies, Vol. 5 (5) , 2014, 6849-6852.

- Vruksha Shah, Riya Patni, Vivek Patani , Rhythm Shah, Prof. Nirmala Shinde, "**Social Media Aggregator using a focused crawler and a web and android UI**", International Journal on Recent and Innovation Trends in Computing and Communication, Version 2,issue 10.

ISSN:0975-9646

Vruksha Shah et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (5) , 2014, 6849-6852

Understanding Focused Crawler

Vruksha Shah, Riya Patni , Vivek Patani, Rhythm Shah

Department of Computer Engineering
K.J.Somaiya College of Engineering
Mumbai, India

Abstract— A basic web crawler can be thought of as a web robot which scans through the web and downloads the pages which can be reached by the links and thus work as an automated program. This leads to a lot of irrelevant information being generated increasing memory overhead. However, a type of crawler which aims to search only the subset of the web related to a specific topic is called a focused crawler. It is comparatively complex but extremely efficient. For predefined topic search, focused crawlers use classifiers and distillers, which help the crawler in collecting the most relevant information. This paper explains the importance of classification and distillation in crawling process.

Keywords— focused crawler, classifier, distiller, crawl, relevant context links.

1. INTRODUCTION

World Wide Web (WWW) contains a large amount of information and every second new information is added such that the size of Web is in the order of tens of billions of pages. To retrieve particular pages from the web, following strategies may be used-

a. Navigate through the web by following the links
b. Search the topic taxonomies and
c. Throw a query using search engine.

Web Crawler is the main component of search engine. It continuously downloads pages and these pages are indexed and stored in database. However, it becomes impossible for a crawler to crawl entire web and keep its index fresh. Thus what one needs is a crawler which aims to search only the subset of the web related to a specific topic. This is called a FOCUSED CRAWLER.[6]

In this paper a survey of different approaches of focused crawling has been described along with importance of classifier and distiller. The outline of this paper is as follows: section 2 describes a brief description of focused crawler and their design issues. Section 3 shows different classification techniques and comparison between these techniques . Section 4 shows how distillation helps to improve the results and in section 5, conclusion is presented.

2. BACKGROUND

2.1. Design Issues

The challenges involved in designing the Focused Crawler are as follows:

- Overloading of websites by the crawler
- Handling large amount of data at any particular time
- Web pages are dynamic in nature
- Crawler should keep a count of how frequently it should revisit a page. (Revisit policy).[2].

So there is a need of a focused crawler which effectively overcomes these design issues and also gives appreciable results.

2.2. Focused Crawler Approaches

A focused crawler can be implemented in various ways.[6] Some of the approaches are shown below:

2.2.1 Priority based focused crawler

In a priority based focused crawler, the retrieved pages are stored in a priority queue instead of a normal queue. The priority is assigned to each page based on a function which uses various factors to score a page. Thus in every iteration, a more relevant page is returned. This is mainly useful in distinguishing between important and unimportant information, wherein priority is given to a more important page.

2.2.2 Structure based focused crawler

Structure base focused crawlers take in account the web page structure when evaluating the page relevance. Its strategy is to compute the relevance score of the page with a predefined formula, then predict the relevance-score of the link, and compute the authority-score of URLs in the queue to be crawled and determine their priority according to the comprehensive value of relevance-score and authority-score namely first crawl relevant and quality page.

2.2.3 Context based focused crawler

Many a times, when a user searches a particular topic on the web, the search system is unaware of the user's needs. For e.g.: If a user is looking for a college university, the search results may include references of that university even on a news portal. Such information becomes irrelevant to the user. This increases the work for the user to filter out unwanted data. To avoid this, we implement a context based focused crawler, which tries to understand the context of the user's needs by interacting with user and comprehending the user profile. The crawler then gets adapted to such contexts and uses them for future search requests.

2.2.4 Learning based focused crawler

Learning based focused crawler is a new learning based approach to improve relevance prediction in focused web crawler. Firstly, training set is built to train the

system. Training set contains value of four relevance attributes:

a. URL word Relevancy
b. Anchor text relevancy
c. Parent page relevancy
d. Surrounding text relevancy.

Secondly ,they train the classifier (NB) using training set. After that trained classifier is used to predict the relevancy of unvisited URL.[4][21]

2.3 Relevancy Calculation Techniques

2.3.1 Weighted Page Rank

In this, weight of web page is calculated on the basis of input and outgoing links and on the basis of weight the importance of page is decided. The relevancy using this technique is less as ranking is based on the calculation of weight of the web page at the time of indexing.[22]

2.3.2 HITS

HITS stands for Hyperlink-Induced Topic Search.

It computes the hubs and authority values of the relevant pages. It gives relevant as well as important page as the result.

When comparing two pages which have received roughly the same number of citations, if one of these journals has received many citations from P1 and P2, which are regarded as important or prestigious pages, this pages needs to be ranked higher. In other words, it is better to receive citations from an important page than from an unimportant one.[22][20]

2.3.3 Eigen Rumor Algorithm

Owing to the increasing number of blogs on the web, it is a challenge to the service providers to display quality blogs to users. When page rank decides rank scores, it gives low scores to blogs and thus such scores cannot be used to decide about the importance of a blog. To overcome this problem, an algorithm was proposed by Fujimura, Inoue and Sugisaki[1] for ranking the blogs. This algorithm called Eigen Rumor Algorithm provides a rank score to every blog by weighting the scores of the hub and authority of the bloggers depending on the calculation of eigen vector.[15][22]

3. CLASSIFIERS.

The most important module of a focused crawler is the Classifier which directly affects the working efficiency of a crawler .Higher accuracy of a classifier leads to higher accurate results. Crawling can be done on a full page content basis or link context basis.

3.1 Types of Classifiers :

3.1.1 Support Vector Machine(SVM)

Support Vector Machines are used to classify data set into distinct classes. It uses a training data set to develop patterns which are represented as points in space. It then uses a mathematical algorithm to assign new examples to specific classes. In SVM, points are mapped such that separate classes are divided distinctly by a clear gap.

3.1.2 Naïve Bayes Classifier

Naïve Bayes classifier is a probabilistic classifier which uses Bayes' Theorem with an independence assumption. It assumes that the presence or absence of a certain feature is independent to that of any other feature. It also incorporates a method of maximum likelihood which estimates the parameters.[25]

3.1.3 Decision Trees based Classification

Decision tree learning is the widespread classification technique. It aims at creating a model that predicts value of a target variable. It first creates a decision tree based on training data set. Once the tree is generated, one simply has to traverse the tree to reach to the leaf and predict a yes or no(In case of Boolean classification). The limitation of decision trees is that it creates a complex model which cannot be generalized well (over fitting) and to overcome this we need to implement pruning.[26]

3.2 Comparing Classification

Pant and Srinivasan [4] compared different classification methods for focused crawling using the full-page content. Their experiment did a comparative analysis between naïve bayes classifier, support vector machines (SVM) and neural network. Naïve Bayes classifier is outperformed by SVM and Neural Network considerably. SVM is better than neural network in the sense that it gets trained faster. The authors suggest that combination of classification methods give better accuracy

Çalışkan and Ozcan[3] implemented a focused crawler using the crawler4j[19], an open source crawler implemented in Java. Their work used the Jsoup library to parse HTML documents. In the experiments, target topic was selected to be the sport news.

The authors used the Weka machine learning library to train topic classifiers. In the initial experiments, Naïve bayes, decision tree (J48 in Weka), and support vector machine are selected. The average length of a news in the dataset was around 20 words.

As the first experiment, the study evaluated naïve bayes, decision tree, and SVM classifiers on the training dataset using 5-folds cross validation. It was seen that while SVM performs the best, Naïve bayes classifier is the worst one. This can be seen from figure 1.

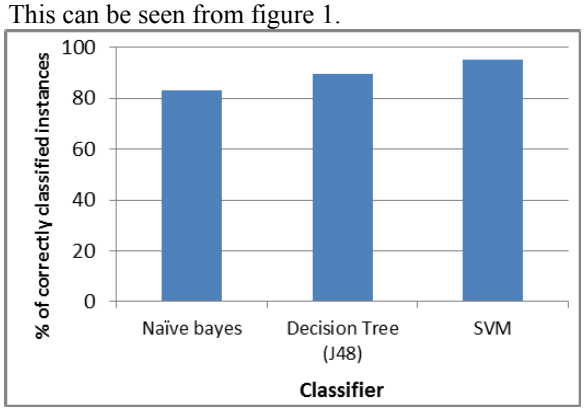


Figure 1

As the second experiment, a different test set was used which consisted of around 10,20 and 40 words link context and the aim was to see the effect of link context size on classifier performance. Figure 2 shows the result of this experiment. Results showed that Support vector machines perform the best among the three. This is shown in figure 2.

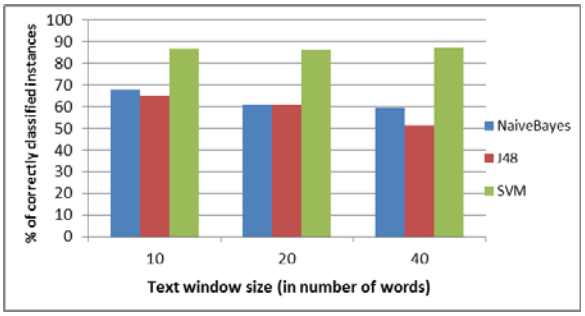


Figure 2

Their experimental results show that SVM classifier performs best compared to Naïve bayes and decision tree classifier. Text window size with 10 words is found as the optimum link context size across different classifiers

4. DISTILLER

Relevance is not the only attribute used to evaluate a page while crawling. Information relevant to the topic but having no outbound links becomes a dead end for a crawler. Here we introduce the concept of Hubs and Authority pages. A good hub page is one that points to many good authorities; a good authority page is one that is pointed to by many good hub pages.

If we talk about social networks, Prestige becomes an important attribute of nodes, especially in the context of academic papers and web documents. Prestige p(u) cannot be solely calculated on the basis of number of back-links. What we need is weighted back-links, that tells us how many important pages point to a particular page. Each node v has two corresponding scores, $h(v)$ and $a(v)$. Then the following iterations are repeated on the edge set E a suitable number of times,

$$a(v) \leftarrow \sum_{(u,v) \in E} h(u) \qquad h(u) \leftarrow \sum_{(u,v) \in E} a(v)$$

Interspersed with scaling the vectors **h** and **a** to unit length. This iteration embodies the circular definition that important hubs point to important authorities and vice versa.

Distillation is not just used as an intermediate component, but it is also an enhancement to the process. In a situation where a highly relevant page is missed out due to improper classification, a distiller comes in handy. For e.g.: a page which contains more images than text is likely to be missed by the crawler(Since crawler mostly relies on textual content). After we use a distiller, we realize that a certain page has a very high prestige p(u), and such a page may then be visited by the crawler. This leads to a more careful retrieval of information. We realize that many of such unvisited links are actually of great importance and worthy of crawling. This can be automated to go parallel with the normal crawling process, thereby saving time and efforts and enhancing the overall performance of the focused crawler. [27]

5. CONCLUSION

A focused crawler is essential for a topic based search. Various types of crawlers are implemented which caters to individual user requirements. Of these, context based focused crawler is useful but difficult to implement, whereas a priority based focused crawler is comparatively easy to implement and is reasonably efficient. Classification is an important step in the crawling process, which can be carried out using three techniques viz. Naïve bayes classification. Decision trees, and support vector machines. Among these, support vector machines prove to be the best compared to the other two with an optimum link size of 10 words. In order to further improve the performance of a crawler, a distiller is used which helps us to re-check whether the chosen page has a high prestige or not, as well to see if any important pages have been missed out, wherein we add such pages to the list of pages to be visited.

6. ACKNOWLEDGMENT

The authors gratefully acknowledge the contributions of Prof. Swati Mali and Prof. Nirmala Shinde and thank her for her endless support and motivation, the college for providing the necessary infrastructure and platform for doing this research.

REFERENCES

[1] Ko Fujimura, Takafumi Inoue and Masayuki Sugisaki,, "The EigenRumor Algorithm for Ranking Blogs", In WWW 2005 2nd Annual Workshop on the Weblogging Ecosystem, 2005.

[2] Swati Mali, B.B.Meshram, "Focused Web Crawler with Page Change Detection Policy" in 2nd International Conference and workshop on Emerging Trends in Technology (ICWET) Proceedings published by International Journal of Computer Applications® (IJCA), 2011

[3] Kamil Caliskan, Rifat Ozcan, "Comparing Classification Methods For Link Context Based Focused Crawlers" , IEEE , 2013

[4] G. PANT AND P. SRINIVASAN, "LINK CONTEXTS IN CLASSIFIER-GUIDED",TOPICAL CRAWLERS," KNOWLEDGE AND DATA ENGINEERING, IEEE,TRANSACTIONS ON , VOL.18, NO.1, PP.107-122, JAN. 2006

[5] I.Partalas, G. Paliouras, and I. Vlahavas. Reinforcement Learning with Classifier Selection for Focused Crawling. In Proceedings of the 2008 conference on ECAI 2008: 18th European Conference on Artificial Intelligence, pp. 759-760, 2008

[6] MEENU,RAKESH BATRA, "A REVIEW OF FOCUSED CRAWLER APPROACHES", IJARCSSE VOLUME 4, ISSUE 7, JULY 2014

[7] S. Chakrabarti, M. Van Den Berg, B. Dom, "Focused Crawling: A New Approach to Topic specific web resource discovery", Proc. Of 8th International WWW conference, Toronto, Canada, May,1999.

[8] Debashis Hati , Amritesh Kumar ," An Approach for Identifying URLs Based on Division Score and Link Score in Focused Crawler" ,International Journal of Computer Applications , Volume 2 – No.3, May 2010.

[9] Jaytrilok Choudhary and Devshri Roy ," A Priority Based Focused Web Crawler" , International Journal of Computer Engineering and Technology , Volume 4 ,Issue 4, july-august 2013.

[10] Sushil Kumar ,Naresh Chauhan , "A Context Model For Focused Web Search", International Journal of Computers & Technology Volume 2 No. 3, June, 2012.

[11] Wenpu Xing and Ali Ghorbani, "Weighted PageRank Algorithm", In proceedings of the 2rd Annual Conference on Communication Networks & Services Research, PP. 305-314, 2004.

[12] Jon Kleinberg, "Authoritative Sources in a Hyperlinked Environment", In Proceedings of the ACM-SIAM Symposium on Discrete Algorithms, 1998.

[13] S. Chakrabarti, B. E. Dom, S. R. Kumar, P. Raghavan, S. Rajagopalan, A. Tomkins, D. Gibson, and J. Kleinberg, "Mining the Web's Link Structure", Computer, 32(8), PP.60–67, 1999.
[14] D. Cohn and H. Chang, "Learning to Probabilistically Identify Authoritative Documents",. In Proceedings of 17th International Conference on Machine Learning, PP. 167–174.Morgan Kaufmann,San Francisco, CA, 2000
[15] Ko Fujimura, Takafumi Inoue and Masayuki Sugisaki,, "The EigenRumor Algorithm for Ranking Blogs", In WWW 2005 2nd.Annual Workshop on the Weblogging Ecosystem, 2005.
[16] E. Gatial, Z. Balogh, M. Laclavik, M. Ciglan, L.Hluchy, "FocusedWeb Crawling Mechanism based on Page Relevance." NAZOU project. Bratislava,Slovakia,2008
[17] Duygu Taylan, Mitat Poyraz, Selim Akyokuş and Murat Can Ganiz, "Intelligent Focused Crawler: Learning which Links toCrawl", IEEE 2011
[18] Sotiris Batsakis, Euripides G.M. Petrakis, Evangelos Milios, "Improving the Performance of Focused WebCrawlers", Data & Knowledge Engineering, Vol: 68, No: 10, pp: 1001-1013, October 2009
[19] HTTP://CODE.GOOGLE.COM/P/CRAWLER4J/
[20] http://en.wikipedia.org/wiki/HITS_algorithm#In_journals
[21] Diligenti, M., Coetzee, F., Lawrence, S., Giles, C., and Gori., M. "Focused Crawling Using Context Graphs.". Proc. 26th International Conference on Very Large
[22] Dilip Kumar Sharma , A.K. Sharma," A Comparative Analysis of Web Page Ranking Algorithms", (IJCSE) International Journal on Computer Science and Engineering Vol. 02, No. 08, 2010, 2670-2676
[23] http://en.wikipedia.org/wiki/Support_vector_machine
[24]https://www.princeton.edu/~achaney/tmve/wiki100k/docs/Naive_Bayes_classifier.html
[25] G. R. Dattatreya and V. V. S. Sarma,"Bayesian and decision tree approaches for pattern recognition including feature measurement costs," IEEE Trans. Pattern Anal. Mach. Intell. vol. PAMI-3, 293-298, (1981).
[26] http://www8.org/w8-papers/5a-search-query/crawling/

International Journal on Recent and Innovation Trends in Computing and Communication
Volume: 2 Issue: 10

ISSN: 2321-8169
3222 – 3225

Social Media Aggregator
Using a focused Crawler and a Web & Android UI

Vivek Patani.
Computer department,
K. J. Somaiya College of Engineering,
Mumbai, India.
vivek.patani@somaiya.edu

Rhythm Shah.
Computer department,
K. J. Somaiya College of Engineering,
Mumbai, India.
rhythm.shah@somaiya.edu

Vruksha Shah.
Computer department,
K. J. Somaiya College of Engineering,
Mumbai, India.
vruksha.shah@somaiya.edu

Riya Patni.
Computer department,
K. J. Somaiya College of Engineering,
Mumbai, India.
riya.patni@somaiya.edu

Nirmala Shinde.
Computer department,
K. J. Somaiya College of Engineering,
Mumbai, India.
nirmala.shinde@somaiya.edu

***Abstract*—**today we live in a digital age, with almost each and every one of us having at least one social media account. With things as trivial as what one had for breakfast to the happenings of the world, everything is discussed about on these social media accounts. This paper intends to suggest using this social media as a policing solution, as in contrast to other traditional media such as television and print media, social media offers velocity, veracity, variety, real-time and a large volume of information. A focused web crawler is used to crawl the internet and to stick to the relevant and required topics only. This crawler is integrated with a database to store the information and the information is projected on an android application as well as a web application for the user's perusal.

***Keywords*—** *aggregator, crawler, focused, social media*

*****

I. INTRODUCTION

The aggregation of relevant information is becoming difficult by the day, as the internet keeps on growing at an astronomical rate. As a result keeping tab of the relevant data and related trends is becoming a tedious task. The solution to this problem is the Social Media Aggregator which collects the relevant data and the related trends in a single portal and subsequently displays it. The data is collected using a focused crawler, following which this data is filtered according to the requirements and stored into a database. The data is then displayed on a Web UI (User Interface) as well as an Android UI, for ease of access.

A. *Motivation*

The data is collected using a focused crawler, following which this data, As the internet consists of myriad amounts of data, to keep tab of trending information from different sources is a tedious and difficult task and hence the motivation to an aggregator which would provide information from various sources, but without testing the authenticity of the information but mostly emphasizing on the relevancy of the information.

B. *Need for the aggregator*

To provide an easy access to the ongoing trend related to a certain domain. This can be better explained by an example, such as; Police can use this in an advantageous way by keeping a tab on the commoners though and can curb flow of wrong information as they are alerted with trends in the society. Sometimes, police can also understand what and how people think and can provide better service by taking into consideration the information provided by the aggregator.

C. *Structure of the paper*

The structure of the paper is based on the flow of technology used and integrated. Primarily the technology and its aspect are discussed which is then followed by explaining what we want to achieve through the social media aggregator. In detail explanation of the aggregator is listed. Then a brief outline of the proposal and its shortcomings are listed.

II. WORKING

In this section, we discuss the working of the aggregator by integrating the underlying technologies.

A. *Working amd architecture of the web crawler*

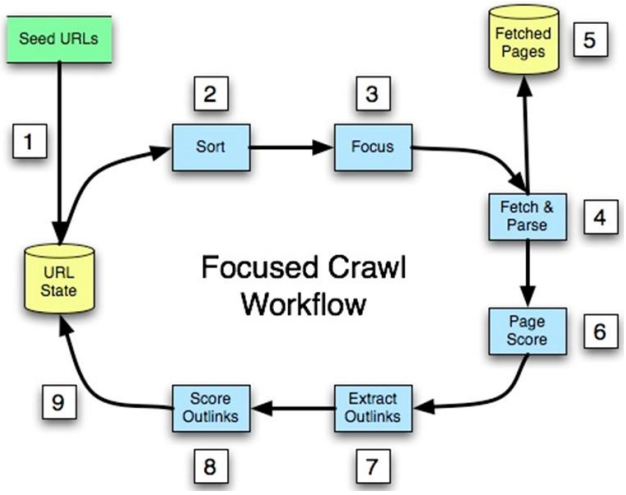


Figure 1: Architecture of Crawler

[5]Initially, the URL State database is loaded with a set of URLs. These URLs can be a broad set of domains with the

highest traffic or the result from some selective searches against some other index or manually handpicked URLs that point tot specific high quality pages. Once the URL state database is loaded, the first loop of the crawler begins. The prime step of all the loops is to extract all the unprocessed URLs and sort them according to their link score. Next is the critical step of deciding which how many URLs to process further in this loop. The fewer the URLs, the tighter the focus of the crawler. The selection can be based on a minimum link score, a fixed percentage of all URLs, a maximum count or a cutoff score that represents the transition point (elbow) in a power curve.

Now, having the set of accepted URLs the fetching process begins which entails all of the usual steps required for polite & efficient fetching, such as robots.txt processing. Pages that are fetched are normally stored in the fetched pages database and are then parsed. The content of the parsed page is given to the page scorer, which returns a value representing how closely the page matches the focus of the crawl. Normally this is a value from 0.0 to 1.0, with higher scores being better. Once the page has been scored, each out-link found in the parse page is extracted and the score for the concerned page is divided among all of its out-links. Finally, the URL State database is updated with the results of fetch attempts (succeeded, failed), all newly discovered URLs are added, and any existing URLs get their link score increased by all matching out-links that were extracted during this loop. At this point the focused crawl can terminate, if sufficient pages of high enough quality (score) have been found, or the next loop can begin. In this manner the crawl proceeds in a depth-first manner, focusing on areas of the web graph where the most high scoring pages are found.

III. Goal Of Aggregator

The Social Media Aggregator uses the Web Crawler to collect information from the internet. When the Web Crawler is first executed it requires the following - 1.) Seed URLs & 2.) Keyword to be searched and tracked. The Crawler provides the user with two kinds of information - 1.) The trending topics & 2.) The sensitive (searched) words with their frequency. The list of sensitive words is already stored prior to running the crawler and is provided by the client as per their requirements. For e.g.:- If a certain company would like to know how much and what is discussed about them and their products in the media and social networks, they would provide the list of sensitive words as their products and Seed URLs as the news portals and social media websites.

A. Proposal & Integration of the Technology

Firstly the java file is executed in order to initiate the process of crawling. Crawler now goes through the various seed URLs and searches for the keywords provided. The data is then inserted into the database through executing SQL Queries which are incorporated in the java file. The log-in credentials of the database are required by the crawler, in order to store the data dynamically. A certain time period is defined for the crawler to run periodically. Generally, one run of the aggregation process on an average consumes 2 hours.

After the crawler completes its iteration and stores the data into the database, the data then needs to be accessed by the Web database and the Android SQLite. The refresh rate of the Android as well as the Web UI needs to be defined; so as to regularly update new data to the intended user, also the user has the option to manually refresh the page.

The proposal of integrating various technologies will result into an outcome as:

- The Word list (Searched or sensitive) and the corresponding frequency of the words.
- Trending topics to a related domain as specified by the user.

IV. Design and Implementation Issues

A. Spam – There are millions of users on social websites such as Facebook, twitter, tumblr etc. and thus have their fair share of spammers. Social spam is unwanted spam content appearing on social networks and any website with user-generated content (comments, chat, etc.).It can be manifested in many ways, including bulk messages, profanity, insults, hate speech, malicious links, fraudulent reviews, fake friends, and personally identifiable information. Hence for its use as a policing solution the aggregator should be able to identify these spam users. For this a duplicate detection algorithm called LSH-with-filtering can be used. It treats the pairs of tweets whose minhash similarities are larger than the threshold as duplicated ones.[7]

B. Duplicate Pages – When the focused crawler crawls the internet it looks for relevant pages and then stores them in a database. If there are duplicate pages the crawler will store them both, hereby increasing the index storage space and he computation cost. The presence of near duplicate web pages thus plays an important role in this performance degradation while integrating data from heterogeneous sources. By introducing efficient methods to detect and remove such documents from the Web not only decreases the computation time but also increases the relevancy of search results. One solution for this is finding these near duplicate pages using minimum weight overlapping method. [8] It uses a TDW matrix based algorithm having three phases- rendering, filtering and verification, which receives an input web page and a threshold in its first phase, prefix filtering and positional filtering to reduce the size of record set in the second phase and returns an optimal set of near duplicate web pages in the verification phase by using Minimum Weight Overlapping (MWO) method.

C. Relevancy of a Page – When deciding whether a page is relevant enough to be stored or not, the crawler checks out the score of the page. However the way these pages are scored is one of the most important aspects of any focused crawler. There are many algorithms available for scoring pages such as genetic algorithm, page rank algorithm, naïve bayes classification algorithm and so on. One such method is classification of links using decision tree induction and neural network classifiers to improve the performance of focused

crawler.[9] Depending upon the requirements and the environment the best fit should be chosen to improve the efficacy of the crawler.

V. TECHNOLOGICAL ASPECTS

This section of the paper discusses the various underlying technologies and their usage with respect to the project at hand.

A. *List of Technologies Used* (though not a complete list but sufficient),

- SQL (Structured Query Language): To create, delete, update entries into the Database.
- Java: To design the Web Crawler.
- MySQL (Server): To store the Data collected by Crawler (Database).
- Eclipse (Java IDE / XML / SQLite): To design the Android Application.
- HTML/CSS/PHP: To design the Web UI and obtain statistics.

B. *Database Storage Aspect*

The web crawler keeps on downloading data and for the storage of this data, a database is essential. The Android UI will make use of the **SQLite** Database - connected to the master **MySQL** which stores data received from the Web Crawler. Queries written will insert the data into the database after applying the necessary filters. The security of data is maintained by SQLAuth as if the data is not protected, it is possible that it could become worthless through ad hoc data manipulation and the data is inadvertently or maliciously modified with incorrect values or deleted entirely. The crawler requires an average of ____GB/MB/KB/B of data. The storage of data follows a particular pattern to maintain a consistency in the Database. Serial Number is maintained as the primary key for accessing records from the Database. We select MySQL as it is open source and also holds good efficiency while input and output of data.

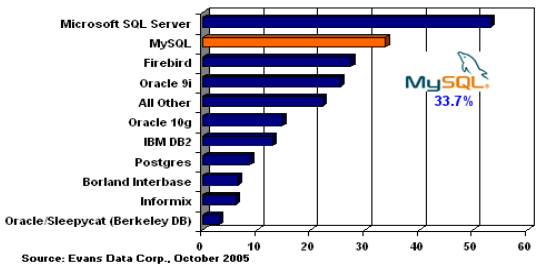


Figure 2: Comparison Chart of Databases

C. *Information Collection Aspect*

The Aggregation of information is based on the concept of a **Web Focused Crawler**. The Web Crawler is designed in the **Java** programming language and majorly consists of commands that can extract information from **HTML** tags. These are helpful in terms of collection of data and also assessing the relevancy of data and filtering the relevant data. The crawler crawls the seed URLs as provided by the programmer and runs up to 'n' levels deep depending upon the requirement. Constraints are applied in order to limit the crawling and to enhance the efficiency, as otherwise it would consume excessive amounts of time and data. The information collected is pertinent to the keywords stored in order to focus our crawling on the related topics and trends only. The frequency of the words is calculated as well and this is taken into consideration while determining the importance of the word and ranking the word on the **trending list**. This list provides us with the **sensitive trending words**. The crawler will for the most part crawl renowned news blogs and various Social Networking websites.

D. *Security Aspects*

Login- ID and password are used in order to gain access to the services. The password is bundled with the **md5 password encryption algorithm**. The android UI is similar as it tends to authenticate user using the same algorithm in order to maintain security of the information. The master database is also secured by the **SQLAuth**. To maintain consistency and exclusive access per user, it is suggested that the mobile number be used as Login - ID and those with exceptional cases should contact the DBA(Database Administrator).

E. *Information Viewing Aspect*

The information that is stored in the database can be viewed by the user in two ways, one is by viewing through the web browser (IE, Mozilla Firefox, Google Chrome or Safari) after logging in to the account & the second method is through the Android application after authorization. There are two different tabs on the Web as well as Android UI to see the list of trends to a related domain and the other tab to view the sensitive word list with frequencies.

VI. RELATED WORK

Due to its astronomical proliferation, social media is receiving a lot of attention now-a-days. The wealth of information it provides can be used for a wide range of application. As such there are number of people working on how best to use the social media. Among the most extensive work is the monitoring of the social network and the analysis of the retrieved information. Semenov, Veijalainen and Boukhanovsky proposed a Generic Architecture for a Social Network Monitoring and Analysis System that consists of three main modules, the crawler, the repository and the analyzer[10]. The first module can be adapted to crawl different sites based on ontology describing the structure of the site. The repository stores the crawled and analyzed persistent data using efficient data structures. It can be implemented using special purpose graph databases and/or object-relational database. The analyzer hosts modules that can be used for various graph and multimedia contents analysis tasks. The results can be again stored to the repository, and so on. All modules can be run concurrently.

Like us Papadopoulos & Kompatsiaris are working on Social Multimedia Crawling for Mining and Search[11] as social multimedia can be leveraged for a wide range of applications, but mining and search systems require innovative crawling solutions to meet both technical and policy-related obstacles.

Also Zhang & Nasraoui have put forth a Profile-Based Focused Crawler for Social Media-Sharing Websites[12] which treat users' profiles as ranking criteria for guiding the crawling process. It divides a user's profile into two parts, an internal part, which comes from the user's own contribution, and an external part, which comes from the user's social contacts. In order to efficiently and effectively extract data from a social media-sharing website for focused crawling, a path string based page-classification method was first developed for identifying list pages, detail pages and profile pages.

VII. ACKNOWLEDGMENT

We would like to thank our guide, Prof. Swati Mali for taking out the time to guide us throughout the project and for helping us as and when required and the K. J. Somaiya College of Engineering, Vidyavihar for providing us with the infrastructure and a platform to help us research and develop such a project.

VIII. REFERENCES

[1] Allen Heydon and Mark Najork, "Mercator: A Scalable, Extensible Web Crawler", Compaq Systems Research Center, 130 Lytton Ave, Palo Alto, CA 94301, 2001.

[2] Francis Crimmins, "Web Crawler Review", Journal of Information Science, Sep.2001. Twitter is the new police scanner http://www.popsci.com/technology/article/2013-04/twitter-is-the-new-police-scanner,2013.

[3] Shi Zhou, Ingemar Cox, Vaclav Petricek, "Characterising Web Site Link Structure", Dept. of Computer Science, University College London, UK, IEEE 2007.

[4] May, 2014: New technical report: "Online Social Media and Police in India: Behavior, Perceptions, Challenges". Authors: Niharika and PK.

[5] http://www.scaleunlimited.com/about/focused-crawler/

[6] Semantic Focused Crawling for Retrieving E-Commerce Information by Huang Wei, Zhang Liyi, Zhang Jidong and Zhu Mingzhu; http://www.researchgate.net/publication/42804620_Semantic_Focused_Crawling_for_Retrieving_E-Commerce_Information

[7] Qunyan Zhang; Haixin Ma; Weining Qian; Aoying Zhou, "Duplicate Detection for Identifying Social Spam in Microblogs," *Big Data (BigData Congress), 2013 IEEE International Congress on* , vol., no., pp.141,148, June 27 2013-July 2 2013

[8] Midhun Mathew, Shine N. Das, "An Efficient Approach for Finding Near Duplicate Web Pages Using Minimum Weight Overlapping Method", *ITNG*, 2012, 2012 Ninth International Conference on Information Technology: New Generations (ITNG), 2012 Ninth International Conference on Information Technology: New Generations (ITNG) 2012, pp. 121-126.

[9] Goyal, D.; Kalra, M., "A novel prediction method of relevancy for focused crawling in topic specific search," *Signal Propagation and Computer Technology (ICSPCT), 2014 International Conference on* , vol., no., pp.257,262, 12-13 July 2014

[10] Semenov, A; Veijalainen, J.; Boukhanovsky, A, "A Generic Architecture for a Social Network Monitoring and Analysis System," *Network-Based Information Systems (NBiS), 2011 14th International Conference on* , vol., no., pp.178,185, 7-9 Sept. 2011

[11] Papadopoulos, S.; Kompatsiaris, Y., "Social Multimedia Crawling for Mining and Search," *Computer* , vol.47, no.5, pp.84,87, May 2014

[12] Zhang, Zhiyong; Nasraoui, O., "Profile-Based Focused Crawler for Social Media-Sharing Websites," *Tools with Artificial Intelligence, 2008. ICTAI '08. 20th IEEE International Conference on* , vol.1, no., pp.317,324, 3-5 Nov. 2008