National College of Ireland

BSc in Computing

2014/2015

Name Surname

Student ID

email

Full Title

Technical Report



**Table of Contents**

[Executive Summary 3](#_Toc441598413)

[1 Introduction 4](#_Toc441598414)

[1.1 Background 4](#_Toc441598415)

[1.2 Aims 4](#_Toc441598416)

[1.3 Technologies 4](#_Toc441598417)

[1.4 Structure 4](#_Toc441598418)

[2 System 5](#_Toc441598419)

[2.1 Requirements 5](#_Toc441598420)

[2.1.1 Functional requirements 5](#_Toc441598421)

[2.1.2 Data requirements 5](#_Toc441598422)

[2.1.3 User requirements 5](#_Toc441598423)

[2.1.4 Environmental requirements 5](#_Toc441598424)

[2.1.5 Usability requirements 5](#_Toc441598425)

[2.2 Design and Architecture 5](#_Toc441598426)

[2.3 Implementation 5](#_Toc441598427)

[2.4 Testing 5](#_Toc441598428)

[2.5 Graphical User Interface (GUI) Layout 6](#_Toc441598429)

[2.6 Customer testing 6](#_Toc441598430)

[2.7 Evaluation 6](#_Toc441598431)

[3 Conclusions 8](#_Toc441598432)

[4 Further development or research 9](#_Toc441598433)

[5 References 10](#_Toc441598434)

[6 Appendix 11](#_Toc441598435)

[6.1 Project Proposal 11](#_Toc441598436)

[6.2 Project Plan 11](#_Toc441598437)

[6.3 Monthly Journals 11](#_Toc441598438)

[6.4 Other Material Used 11](#_Toc441598439)

# Executive Summary

The purpose of this document is to set out the requirements for the development of a Search Engine for News. The Search Engine will bring up reliable news sites/articles. As we are trying to exclude fake news and the sites that provide these fake articles. The search engine should be optimized to exclude all fake news outlets and their articles.

The intended customers are people who keep up to date on current events/ worldwide news. Eliminating fake news from their feeds. The Search Engine will be created using different languages such as Java, Javascript, C++, Python and MySql, Service API’s will also be used.

I have always been interested in the likes of Google and Yahoo, how they work and the algorithms used to create software that is simple to use. Using the likes of Google and Yahoo is very easy, anyone could use it. But it is the background of this that I am interested in the most. I have worked in different aspects of IT, from Web Development, IT Support and a Software Tester. The thing that interested me the most was the problem solving being done to create different types of software, and from reading up on Google’s algorithm and seeing the amount that went into making Google Search amazed me.

But one thing that can be very irritating on searches is the amount of Fake Websites that appear, websites dishing out false news stories and headlines to receive views on their sites. I would like to try an remove these websites from people’s searches and provide a better, trusted source for their news.

# Introduction

This template for technical report is provided for your convenience. It should be seen as a guide rather than an obligatory form. Your individual report might require changes in terms of format or content (i.e., headings) or both.

Print on one side of the paper only (this will be the right hand side when the pages are bound).

## Background

After talking with Michael about particular idea’s for my project, he expanded on one of the ideas. Originally my idea was to look into the social media aspect of things, creating reusable API’s that could gather data from multiple social media platforms. I thought this would be useful for people who were researching various different social media platforms, Snapchat, Pintrest and Linked for example.

Michael then suggested about fake news as this appears a lot on social media platforms, fake accounts, previous popular accounts that have been purchased to spam the user’s news feed with these fake articles. At first I thought I could investigate this through Social Media’s trying to come up with an algorithm to detect fake accounts that could be spamming news feeds. But the resources would not be available. The next best thing was to create an application that could rank News articles, Websites and Piece Author’s by Prestige. Hopefully this could eradicate the fake news websites from appearing in any search carried out.

## Aims

The overall aim here is to rank News Websites, Articles and Authors by their prestige. If we use the Irish Independent as an example; A very trusted News distributer in Ireland. Compare this to TheLADBible or Joe.ie. The likes of Joe.ie and TheLADBible would not be as reliable as the Independent. These are relatively new websites that provide the users of Social Media with news and reports on sports, events and many other lines of news. But they can also spam news feeds with articles coming from untrusted sources, their articles would not be as reliable as that coming from a fulltime journalist working for the likes of the Independent. The end goal is to take out webpages of news distributers like the two prior examples. People want real news stories and not to be wasting their time reading

## Technologies

How? - Brief description of the technologies used in the project. Do not copy & paste descriptions from websites here, but describe what it is and how it contributes to your project.

## Structure

Brief overview of each chapter

# System

## Requirements

Experienced controllers shall be able to use all the system functions after a total of one hour training session. After this training, the average number of errors made by experienced users shall not exceed two per day.

### Functional requirements

This section will give an overview of the functional requirements, each requirement is provided with a small summary and each requirement into its use cases.

1. Register new user:

The application will allow any potential new user to setup a profile so that they can access all of the necessary features.

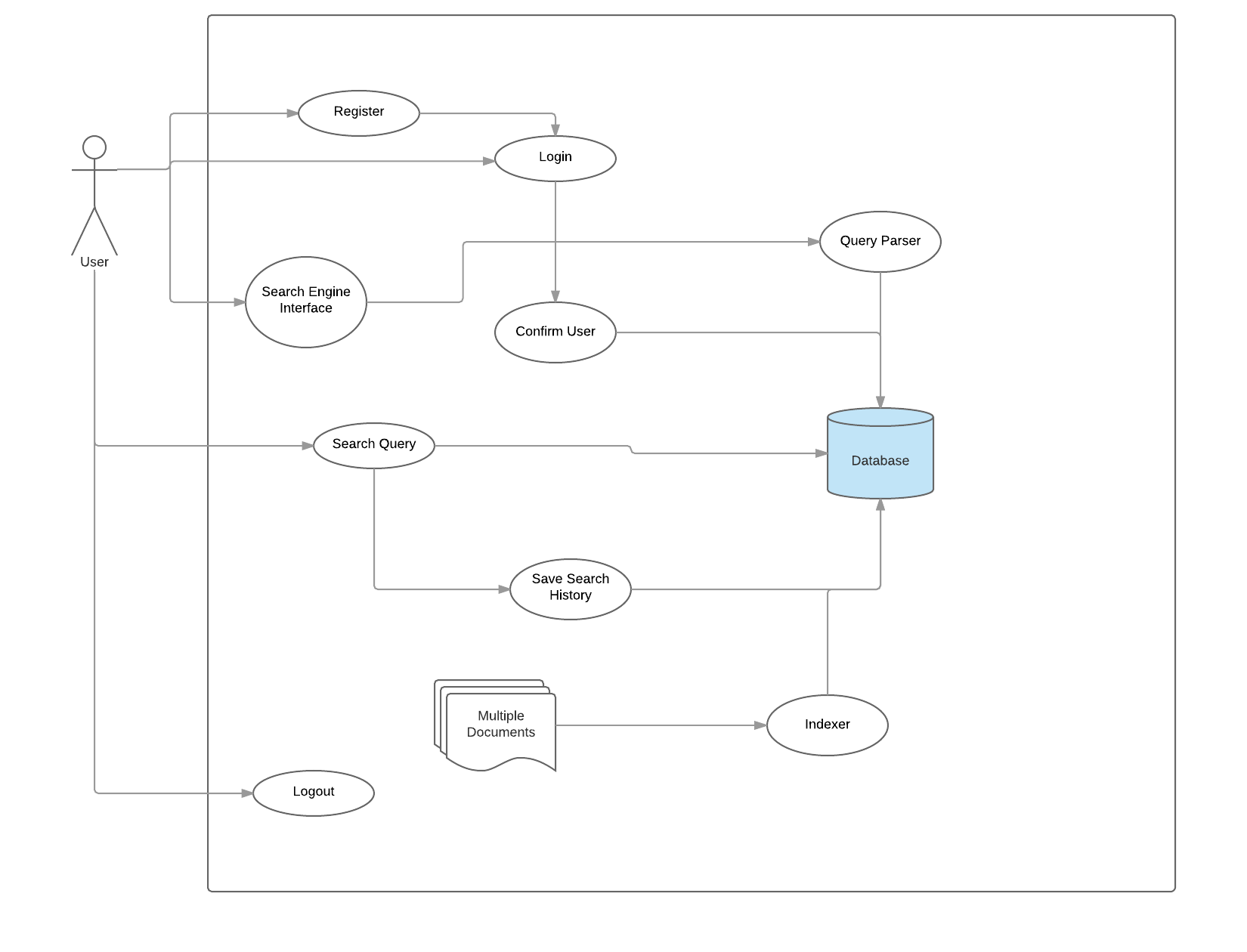
1. Login / Logout:

A user can login and logout when they need to, that will end their session with the app and upon logging back in will be back at the start screen. This will prevent confusion between different users.

1. Offline Tasks: Much of the processing is done before.
2. Crawling – The Webpages are crawled from the web.
3. Indexing – Documents are read and an index is created and kept on a disk. Containing the exact location of the webpages, storing them in the Index. When called upon will need to be shown quickly.
4. Online Tasks: Processing is done in real time. Once the user enters their query into the Engine.
5. Query Parser – Parses the query and processes it. Converting the query into lowercase and removing words from the query and stemming. The engine uses the query tokens generated after processing to search the pages from the index.
6. Page Ranker – After retrieving the list of documents/webpages containing the query term, this will rank the documents/webpages such as the most relevant page or the best page.

### Use Case Diagrams

**Full Use Case Diagram**



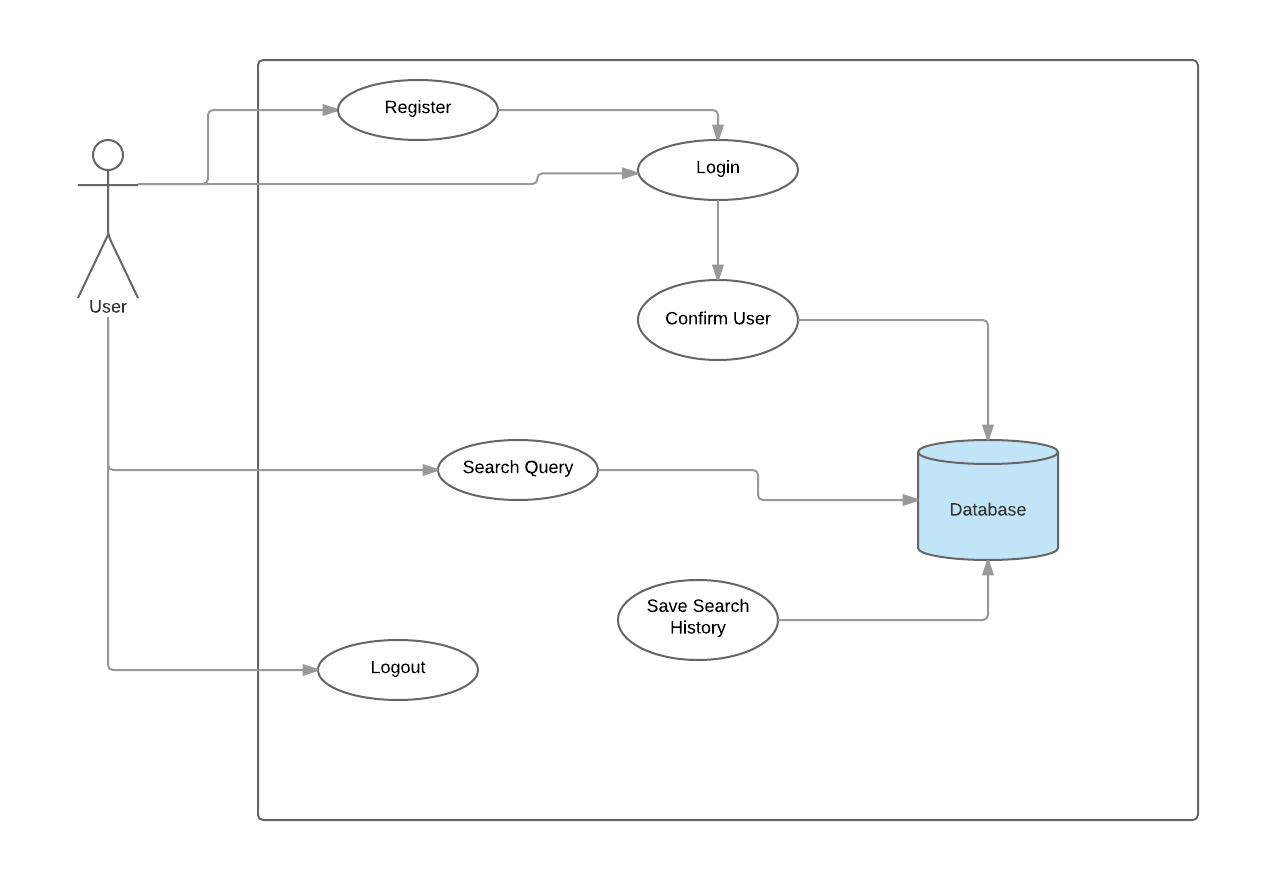
**User Log in Use Case Diagram**

**Scope**

The user’s interaction once opening the application

**Actors**

1. User
2. Database



**Description**

**Requirement**

For the user to get to this stage they must be registered

**Activation of Application**

Active upon launching of application by a registered user.

**Main flow**

1. User Launches Engine
2. The User registers or enters their credentials
3. The system saves the new user credentials to Database.
4. User can then Search
5. Search History will be saved to the Database

**Extraordinary flow**

The user decides to exit the application before the registration process has been completed, not allowing the user to be registered and therefore not creating a new user.

**Discontinuation**

Application closed by the potential new user.

**After Discontinuation condition**

The system goes into a waiting state until it receives new interaction from the user.

**Offline and Online Tasks Use Case Diagram**

**Description & Priority**

This can only be completed if the user has previously created an account successfully, once they have logged in they then will be presented with the Home screen and will be able to also log out therefore ending the session.

**Scope**

The user’s interaction with the android application upon opening it for the first time

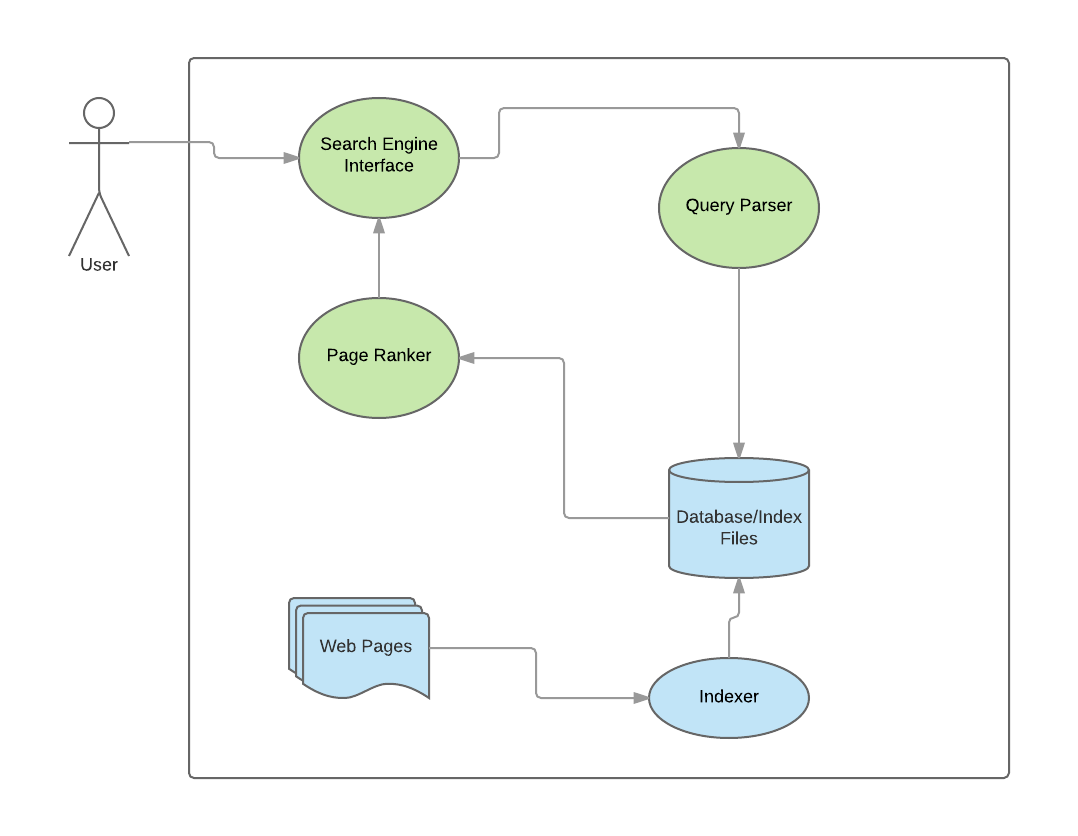
**Actors**

1.     User

2.     Database

**Use Case Diagram**

The following diagram displays a user logged in preforming a search.



**Description**

**Offline Tasks:**

1. Web Pages
2. Indexer

**Online Tasks:**

1. Search Engine Interface
2. Query Parser
3. Page Ranker

**Requirement**

For the user to get to this stage they must be registered and logged in

**Activation of Application**

Active upon launching of application by a registered user.

**Main flow**

1. User Launches Engine
2. The User registers or enters their credentials
3. User Searches a Query
4. Raw Query is sent to the query parser
5. Clean and processed query tokens are searched in the Index
   1. Read the webpages and make index
   2. Write index files to disk
6. Ranked list of items are retrieved

**Extraordinary flow**

The user decides to exit the application before the registration process has been completed, not allowing the user to be registered and therefore not creating a new user.

**Discontinuation**

Application closed by the potential new user.

**After Discontinuation condition**

The system goes into a waiting state until it receives new interaction from the user.

### User requirements

1. Simplicity

The Search Engine should be as easy to use as Google. Using a search function for News from around the world. Real news from around the world will be provided as End Users do not want to read Fake News articles. The simple interface and use is extremely important.

The Search Engine will be optimized to show news outlets websites from around the world. Articles will be organized by category. The Search function will pick up what the user is looking for and will output news related to their search.

1. Security

A possible Login Feature may be added. User will need to provide:

1. First Name
2. Surname
3. Email Address

### Environmental requirements

1. Storage: The database must be able to store all of the registered users and all of their saved searches alongside the Indexes.
2. Cellular: The application should be able to work on mobile devices using cellular networks.

### Usability requirements

## Design and Architecture

Describe the design, system architecture and components used. Describe the main algorithms used in the project. (Note use standard mathematical notations if applicable).

An architecture diagram may be useful. In case of a distributed system, it may be useful to describe functions and/or data structures in each component separately.

## Implementation

Describe the main algorithms/classes/functions used in the code. Consider to show and explain interesting code snippets where appropriate.

## Graphical User Interface (GUI) Layout

Provide screenshots of key screens and explain.

## Testing

Describe any testing tools, test plans and test specifications used in the project

## Customer testing

Provide evidence for and results of customer testing. This may include ratings or quotes from the customer.

## Evaluation

How was the system evaluated and what are the results? In many cases this will include usage data and user feedback. It may also include performance evaluations, scalability, correctness, etc. depending on the focus of the project.

Quantative results may be reported in tables or figures. Note that tables have their caption above the table and need to be cross referenced in the text (see **Error! Reference source not found.**). In many cases, tables are better to read if you skip the vertical lines.

Table 1: Performance with and without caching

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Nwithout** | **Nwith** | **Std.-Deviationwith** | **Std.-Deviationwithout** | **p** |
| Records | 100 | 200 | 2.54 | 3.97 | .002 |
| Data (GB) | 100 | 200 | 2.54 | 3.97 | .002 |
| Speed | 100 | 200 | 2.54 | 3.97 | .002 |

Figures have their caption below the figure as shown in **Error! Reference source not found.**. Make sure that if you use colour, the figure is still readable when printed in black & white, e.g., by using additional symbols, patterns, etc.



Figure 1: Learning gain across different experimental groups

# Conclusions

Describe the advantages/disadvantages, opportunities and limits of the project.

# Further development or research

With more resources, where could the results of this project lead to?

# References

It is recommended that students use the APA, Berkeley, Harvard or other internationally approved style. Here is an example of the APA citation style:

Wilcox, R. V. (1991). Shifting roles and synthetic women in Star Trek: The Next Generation. *Studies in Popular Culture, 13*(2), 53-65.

In the text this article can be cited as “Wilcox (1991)” or “(Wilkox, 1991)”.

References to web sites must include the access dates.

The library provides a study guide on Harvard style referencing.

# Appendix

## Project Proposal

## Project Plan

## Monthly Journals

## Other Material Used

Any other reference material used in the project for example evaluation surveys etc.