Martello.io

Technical Report

Web application utilising Web Mining and Natural Language Processing techniques to monitor, analyse, and interpret news articles and social media posts for journalists and other media professionals

Adam O’Callaghan
BSc in Computing
x13116525
adam.ocallaghan@student.ncirl.ie

National College of Ireland

CONTENTS

EXECUTIVE SUMMARY

As the massive amounts of both structured and unstructured data produced daily continues to grow, it is evident that keeping abreast of current events, ideas, and trends is becoming increasingly difficult. Many press officers, journalists and media professionals resort to the old method of combing through websites, print media, and other content streams in order to get an idea of what is currently happening. However, this scattershot approach is far from optimal as key stories and information can often slip past the person’s attention. Complicating matters is non-stop, 24-hour nature of the global news cycle, whereby stories appear and disappear in quick succession. All of these factors make it close to impossible for any one person to stay aware of each breaking news story.

However, as artificial intelligence technologies continue to mature, new techniques are appearing which allow computers to track, extract, and analyse the vast array of information that is produced each day. One of these technologies makes use of a set of algorithms known as Natural Language Processing, which allows computers to analyse the content of natural human language and derive new information from it.

The application of these Natural Language Processing algorithms to online articles, news, and other media content is the central concern of the Martello.io software system. Martello will actively scrape content from the websites and newsfeeds of the top media outlets and process this information using a number of algorithms. The results of this NLP processing will allow the output of summaries and categorisation, the identification of named people, places and organisations, and the detection of the sentiment of the articles themselves in order to see if it is a positive or negative opinion.

Customers using Martello will be able to make use of a full-featured application that allows them to quickly extract information pertinent to their organisations and narrow the vast quantity of news articles down by location, category, date, or by specific people, companies and industries. Additionally, customers will be able to view the positive or negative sentiment of relevant news articles, or collections of articles, in order to quickly ascertain the prevailing sentiment about specific subjects and topics.

INTRODUCTION

BACKGROUND

As the quantity of news content, articles and other media produced each day online grows, press officers, journalists and other media professionals across various sectors find it increasing difficult to stay abreast of news critical to their roles. Many rely on manually combing through relevant news sources each morning, while others use third-party companies which essentially outsources this for them – neither solution being optimal.

Complicating matters is the nature of the 24 hour news cycle, whereby it's difficult for individuals to keep up with the sheer quantity of news being produced on a daily basis, and the various social media platforms which essentially produce constant streams of new content, opinions and consumer sentiment.

These factors combine to create a fast-moving and constantly-changing media environment for journalists, press officers and brand representatives, with the current state of things often being elusive and hard to nail down.

IDEA

With the increase in processing power and constantly improving algorithms, computers are getting progressively better at processing unstructured text, with many libraries across multiple programming languages now able to parse text, social media posts, html pages, and various other sources of natural language to infer meaning, sentiment and many other attributes previously only possible by human means.

Through Natural Language Processing (NLP) libraries it is possible for computers to carry out much of the preliminary work that once fell to journalists and other media professionals. Analysing news stories across multiple media sources in real time can be done much faster by NLP algorithms than by individual journalists.

With the above in mind, I am proposing to develop a cloud-based web application that monitors and scrapes news articles and posts from the top news and social media outlets online, and then processes these articles using Python's Natural Language Toolkit (NLTK) in order to determine:

- Named Entities (e.g. Enda Kenny, National College of Ireland, Microsoft, etc)
- Sentiment (positive or negative intent behind the text)
- Categories
- Summaries

The processed, and now structured, information will then be deposited into a database that is accessible from a web application frontend by the end user.

TECHNOLOGIES & APPLICATION

WEB-SCRAPING

A directory of online media and news outlets will be compiled in order to find the most relevant and influential in each locale. Outlets that expose their data publicly via an API will be queried for the information required at the Natural Language Processing stage. As many sites will not have APIs implemented the news or RSS feeds of these websites will be scraped using Scrapy, a Python web-scraping library.

SOCIAL MEDIA APIS

The APIs of the main social media platforms will be used to obtain posts, articles, microposts and comments which will then be further processed using NLP. This aspect of the application will complement the previous scraping and analysing of news articles, in that it will identify the level of social engagement of topics and how they are being perceived by the general public (as opposed to professional journalists).

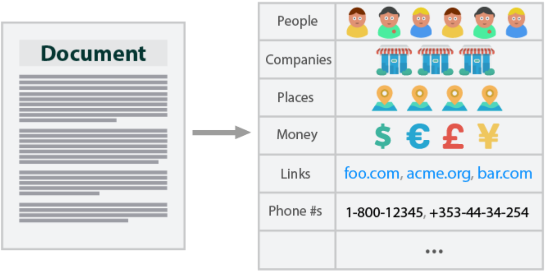
NLP PROCESSING

The retrieved news and media outlet information will then be stored in a MongoDB database before it is processed using Python's Natural Language Toolkit (NLTK). Social media content will be passed directly through NLTK and into the web-application database.

NLTK will process the content for various attributes and meaning. It is also worth noting that the below listed attributes are a small subset of the many possible factors that NLTK can identify in a text corpus, and that many others may be of use or interest in this application.

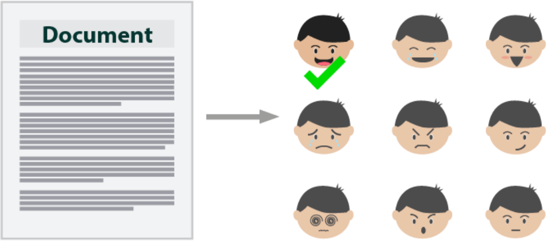
NAMED ENTITY RECOGNITION

Named Entity Recognition identifies proper names from the scraped information and will allow the end user to filter by specific people, organisations and locations.



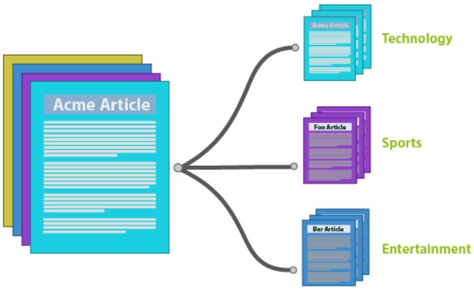
SENTIMENT ANALYSIS

Sentiment analysis gives a rating as to level of positive or negative sentiment behind a piece of text, allowing users to quickly observe the prevailing thought about certain topics, subjects or people.



CATEGORISATION

Categorisation identifies pieces of text as falling into predefined categories, allowing the end user to search according to specific tags and categories in order to narrow down the volume of information.



SUMMARISATION

Summarisation condenses a text down to its most important sentences and information, allowing the end user to quickly comprehend the substance of an article without having to read through the entire content.

WEB APPLICATION

NLTK will pass the processed information (now in a structured format) into a MongoDB database which is directly connected to the web application. The processed social media information will also be stored in a MongoDB database. The application itself will be built using NodeJS for the server and BlazeJS for the frontend (on top of the ExpressJS framework).

RESOURCES REQUIRED

The programming languages and technologies used to build Martello.io are largely free and open-source, with the only expenses coming from web and database hosting, and domain registration. The following resources will be required throughout the project.

Resource Type	Resources
Programming Languages:	JavaScript
Frontend:	HTML5 and CSS3
Backend:	NodeJS and BlazeJS
Pre-Processing:	Meteor-Scrape, NLP-Compromise, and NPM-Sentiment
Web App Hosting:	Heroku
Database Hosting:	MongoLab
Domain Registration:	GoDaddy

MARKET

As this application will be built using some of the latest technologies it will be used for a number of novel and interesting tasks. Some of these are listed below, but it is envisioned that as the application matures and evolves it will be utilised for purposes well beyond the scope of this document.

POTENTIAL USE CASES

PRESS OFFICE

In many organisations the Press Office needs to stay informed of any news articles pertaining to specific individuals. Using the web application proposed in this document, staff members will be able to save the names of these people as keywords in the system, quickly able to retrieve and view all the latest news articles about that person.

An example of this could be that the Press Office at the Department of Transport might want to save keyphrases such as "Shane Ross", "Minister for Transport", "Dublin Bus Strike", etc, in order to quickly find news stories about those keyphrases. Staff members could then look over the returned article summaries, as well as the sentiment analysis, in order to see the prevailing mood towards each in the media.

JOURNALIST

While Press Offices may use the application for observation and reaction, journalists themselves could also use it for research and sourcing of new article subjects and topics.

For example, a journalist at a smaller media outlet would not have the on-the-ground resources of Reuters, the Associated Press or the BBC in the case of breaking news stories from conflict zones. Rather than having to search through various – often conflicting – news sources to ascertain what is happening, the journalist could use this application to quickly source information by viewing summaries of the critical information in each article, in turn allowing them to have a quicker turnaround for their own story.

COMPETITORS

NEWSWHIP

Newswhip are an Irish company focusing on online content discovery and social engagement of news stories. While Newswhip firmly exists in the news and social media space they seem to operate predominantly on discovery of stories in any area that have the potential for high social velocity and virality online, as opposed to the monitoring of all stories pertaining to specific companies, industries or people.

AYLIEN

Aylien is another Irish-based company that is a provider for Natural Language Processing APIs. While operating as a vendor for NLP and Machine Learning algorithms and tools, Aylien itself does not have its own web application harnessing and utilising these algorithms, and instead focuses on selling these tools to developers in order for them to create their own applications.

KANTAR MEDIA

Kantar Media are an international company with a presence in a number of markets. Kantar's *Media Monitoring* solution most closely aligns with the functionality of the proposed application; however, Kantar also incorporate a human element into their solution which increases costs and adds additional time to the analysis, both of which Martello.io is intended to minimise when in production.

REQUIREMENTS

INTRODUCTION

PURPOSE

The purpose of this section is to set out the requirements for the development of *Martello* – a cloud-based, Software-as-a-Service web application utilising Web Mining and Natural Language Processing techniques to monitor, analyse, and interpret news articles and social media posts.

The intended customers for Martello are journalists and other media professionals – including, but not limited to, PR consultants, press officers, and marketing managers – whose job entails as a core component that they stay abreast of breaking news important to their companies and organisations.

PROJECT SCOPE

The scope of the project is to develop a cloud-based web application that monitors and scrapes news articles and posts from the top news and social media outlets online, and then processes these articles using natural language processing algorithms in order to determine core features - such as named entities, article sentiment, categories, and summaries – that could be of value to organisations. The processed, structured information will then be stored in a database that is accessible from a web application frontend by the end user.

The system shall consist of two main architectural components – a pre-processing stage, where web mining, natural language processing, and data cleaning will take place; and a web application, whereby end-users will be able to log in and utilise the data collected to discover previously unknown business intelligence.

Adam O'Callaghan was involved in discussions with the Press Officer at the *Department of Arts, Heritage, and Gaeltacht Affairs* to elicit the following requirements.

DEFINITIONS, ACRONYMS, AND ABBREVIATIONS

The following definitions, acronyms and abbreviations apply to this project...

Acronym / Abbrev.	Name	Definition
NLP	Natural Language Processing	The use of algorithms written specifically for the processing of text-based information
NLTK	Natural Language Toolkit	A set of Natural Language Processing algorithms written in Python
MongoDB	MongoDB	A NoSQL database that uses Javascript as it's query language
Node	Node.js	A Javascript server framework for the development of fast, highly scalable web applications

Blaze	Blaze.js	A Javascript library for building frontend interfaces
Heroku	Heroku	Scalable hosting platform
API	Application Programming Interface	Software interfaces used to access the data and resources of external software systems
Web-scraping	Web-scraping	Used to retrieve information and data from websites by extracting it from the rendered HTML

USER REQUIREMENTS DEFINITION

Customers using Martello will require a full-featured application that allows them to quickly narrow down and extract information pertinent to their organisations. Narrowing the vast quantity of news articles down by location, category, date, or by specific people, companies and industries named is a core feature that will have to be implemented Martello.

Additionally, customers will need to be able to view the positive or negative sentiment of relevant news articles, or collections of articles, in order to quickly ascertain the prevailing sentiment about specific subjects and topics. This is required in order to allow media professionals to act and respond in a prompt manner to emerging stories.

Martello will need to be very intuitive to use from the customers point of view – the function of every part of the interface should be easy to figure out at a glance, without the need for overly wordy explanations of the various filter and widgets available. The application will have to respond instantly to filters, with the current information near-instantaneously appearing onscreen.

As can be seen from the above, the ability to quickly narrow down lots of content into easily digestible, yet critically important, pieces of information is a central aspect of the system. Speed is another area on which the system will have to perform well on – breaking news stories are just that: *breaking*; and as such, Martello will be required to facilitate media professionals in rapidly responding to stories and emerging trends in the media.

REQUIREMENTS SPECIFICATION

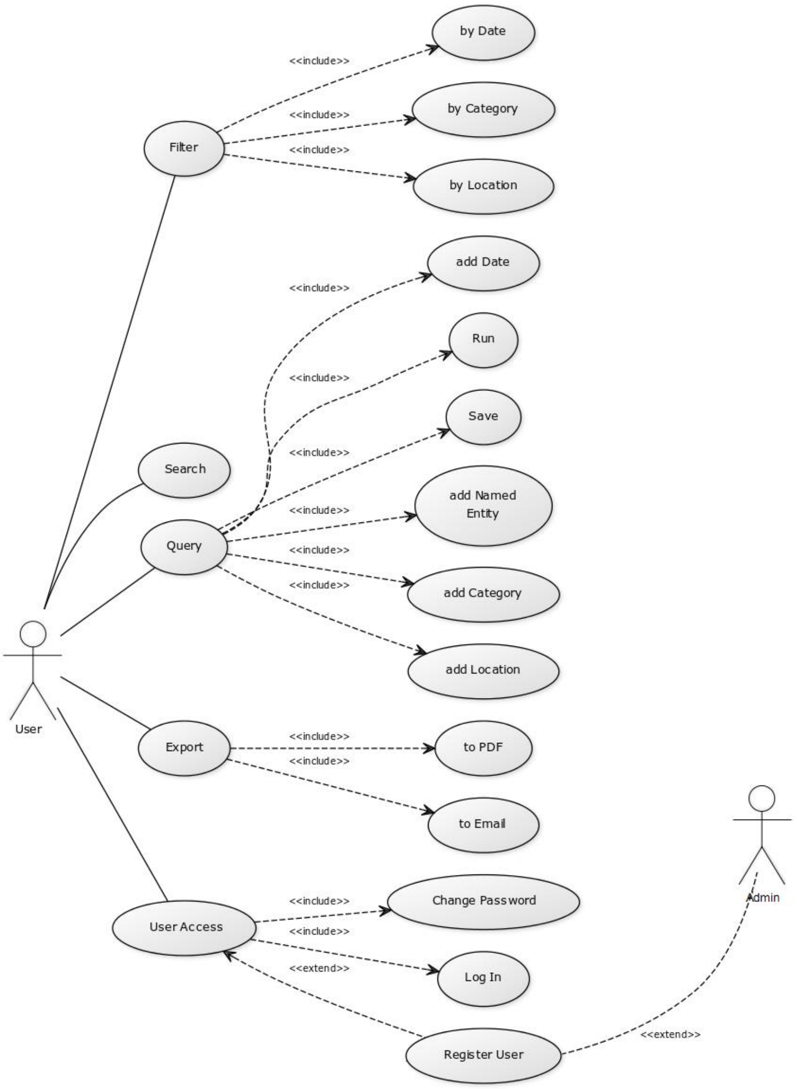
FUNCTIONAL REQUIREMENTS

The following list contains the functional requirements that will be implemented for the Martello system. Each requirement has a priority level as set out in the following table…

Priority Level	**Priority Description**
Priority 1	Highest Priority – System Critical
Priority 2	High Priority – Core Functionality
Priority 3	Mid Priority – Main Functionality
Priority 4	Low Priority – Requested Functionality

Priority 5	Lowest Priority – Nice to Haves

USE CASE DIAGRAM



REQUIREMENT 1: USER ACCESS

DESCRIPTION & PRIORITY

As Martello is a Software-as-a-Service application, user login is restricted to customers who have already been registered by a Martello administrator. Once registered by an admin the user will have instant access to their account. Accessing the system *Priority Level 1* as users cannot perform any action unless they are logged into Martello.

USE CASE

SCOPE

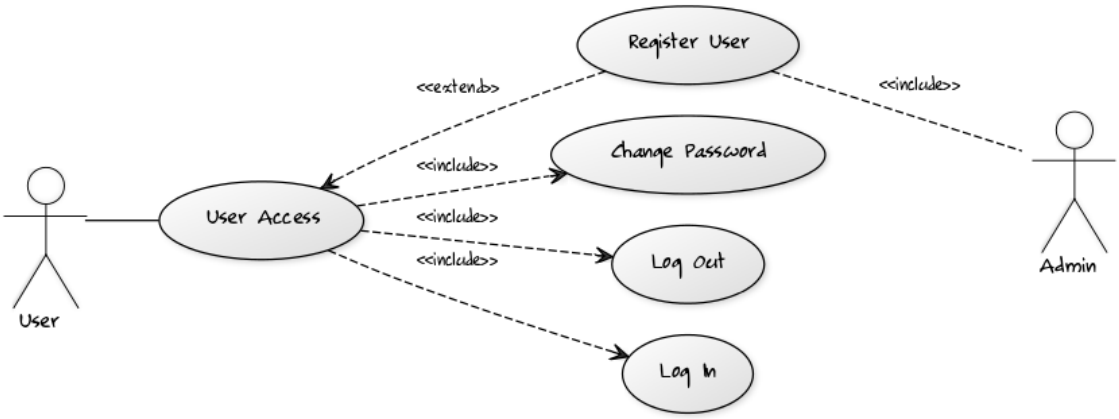
The scope of this use case is to control for non-administrative Martello users accessing the software system and the changes they can do relating to system access.

DESCRIPTION

The use case describes how a User logs in, changes passwords, and logs out. Registration of users is on the Martello system is done by Admin.

USE CASE DIAGRAM

The following diagram outlines the actors and uses cases involved in User Access requirement...



FLOW DESCRIPTION

PRECONDITION

The user must be registered to use the system by an Admin.

ACTIVATION

This use case starts when the User attempts to log into the system.

MAIN FLOW

1. The system identifies the User as either Registered or Not Registered
2. If the User is Not Registered (See A1)
3. If the User is Registered but has forgotten their password (See E1)
4. The User is now logged in and can access the system
5. The User attempts to Log Out

ALTERNATE FLOW

A1: Not Registered

1. The system responds by asking for the User's email address and to specify a password
2. The User enters their email address and submits
3. An Admin registers the User with the system
4. The User receives their logon details via email and attempts to log into the system
5. The use case continues at position 4 of the main flow

EXCEPTIONAL FLOW

E1: Forgotten Password

1. The system responds by asking for the User's email address
2. The User enters their email address and submits
3. An System resets the User's password to a randomly generated one
4. The User receives their new password via email and attempts to log into the system
5. The use case continues at position 4 of the main flow

TERMINATION

The User is now logged out and cannot access the system.

POST CONDITION

The system remains available for the User to access again at any point via the logon portal.

REQUIREMENT 2: FILTER

DESCRIPTION & PRIORITY

A core function of Martello is being able to narrow down vast quantities of news articles and information into easily digestible chunks. The main way in which this is done is through the use of filters. Martello will collate and process a mass of online data and info, but narrowing it down will be at the discretion of the users of the system. They will be able to narrow easily by location, category, or date as a first port-of-call in the filtering process. Filtering is *Priority Level 2* as it is a core functionality of the system.

USE CASE

SCOPE

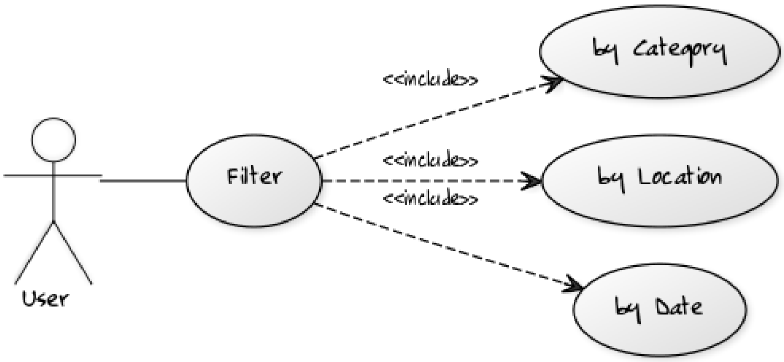
The scope of this use case is how Martello lets users to narrow down the large amount of media information that is stored in the system through the use of filters on the main dashboard.

DESCRIPTION

The use case describes how a User can filter by location, category, and date while on the main dashboard of Martello.

USE CASE DIAGRAM

The following diagram outlines the actors and uses cases involved in Filter requirement...



FLOW DESCRIPTION

PRECONDITION

The user must be logged into the system.

ACTIVATION

This use case starts when the User has logged into the system and wants to manipulate data on the main dashboard.

MAIN FLOW

1. User selects a Location, Date, or Category filter
2. The System applies the selected filter to the data
3. The dashboard interface updates
4. User selects a secondary filter (See A1)
5. User selects a tertiary filter (See A1)
6. User selects the Reset Filters option (See E1)

ALTERNATE FLOW

A1: Apply Additional Filter

1. The System applies the selected filter to the data on top of the previously applied filter
2. The dashboard interface updates with narrowed data

EXCEPTIONAL FLOW

E1: Reset Filters

1. The System clears all applied filters
2. The dashboard interface updates to its original state

TERMINATION

The User either clears all filters or exits the system entirely.

POST CONDITION

The filter options remain accessible by the User to reapply to the data.

REQUIREMENT 3: QUERIES

DESCRIPTION & PRIORITY

The ability to filter is the core functionality of Martello, but filtering on its own means that users have to manually apply their specified filters to the data in order to make use of the system. The ability to set up stored queries that can be applied in a single click is another functional requirement of the system that customers have said would be beneficial to the system. This requirement is *Priority Level 3* as while it is a main component of Martello the same information can be gathered through other means.

USE CASE

SCOPE

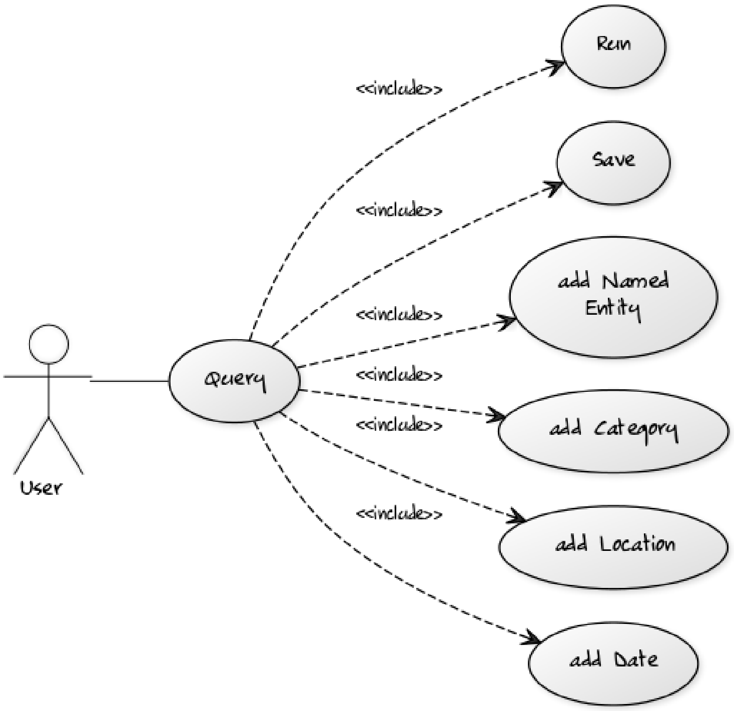
The scope of this use case is the creation and application of queries that narrow data down to commonly used sets of filters in a single click of a button.

DESCRIPTION

This use case describes how a User can create a query, add filters to it, name the query, and then apply that query in a single click whenever it is required.

USE CASE DIAGRAM

The following diagram outlines the actors and uses cases involved in Queries requirement...



FLOW DESCRIPTION

PRECONDITION

The user must be logged into the system.

ACTIVATION

This use case starts when the User has logged into the system and wants create, store or run queries to quickly apply multiple filters from the dashboard.

MAIN FLOW

1. User is on the dashboard
2. User selects a named Query from the stored queries (See A1)
3. User selects the Create Query option
4. User selects a Location and/or Date and/or Category and/or Named Entity filter
5. User gives the Query a unique name
6. User clicks on Save
7. The System creates the Queries
8. The System displays the Query in the Queries options on the dashboard

ALTERNATE FLOW

A1: Run Query

1. The System applies the selected query to the data
2. The dashboard interface updates with narrowed data
3. User selects the Reset Filters option (See E1)

EXCEPTIONAL FLOW

E1: Reset Filters

3. The System clears all applied filters
4. The dashboard interface updates to its original state

TERMINATION

The User either clears all filters or exits the system entirely.

POST CONDITION

The query options and stored queries remain accessible by the User to reapply to the data.

REQUIREMENT 4: EXPORT

DESCRIPTION & PRIORITY

After users have either created and run queries, or simply applied a filter (or filters) on the data, they may want to export this information outside of Martello for use or review by themselves or others in their organisation. This requirement is *Priority Level 3* as while it is a main component of Martello all the information can be displayed and utilised without the need to export it from the system.

USE CASE

SCOPE

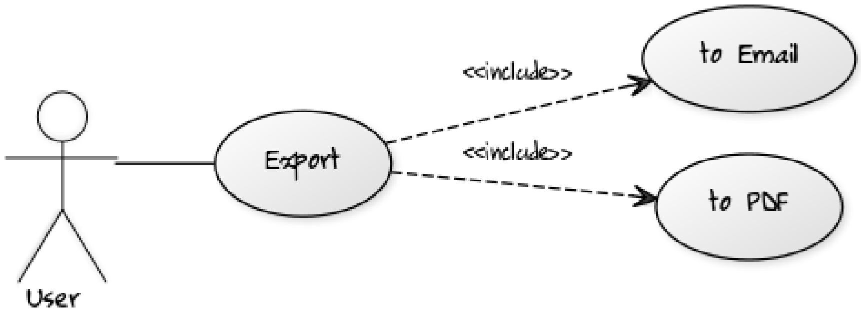
The scope of this use case is how Martello users can export their narrowed information from the system in order to use it outside the system.

DESCRIPTION

The use case describes how a User can export their selected information to either PDF format or email and send that information outside the system.

USE CASE DIAGRAM

The following diagram outlines the actors and uses cases involved in Export requirement...



FLOW DESCRIPTION

PRECONDITION

The user must be logged into the system and have either run a query or set a filter, or multiple filters, on the data.

ACTIVATION

This use case starts when the User has logged into the system, run a query or applied filters, and now wants to export the data for use or review outside of the system.

MAIN FLOW

1. User runs a query or applies filters
2. The System applies the selected filter to the data
3. The dashboard interface updates
4. User selects the Export option
5. User selects the To PDF option (See A1);
6. User selects the To Email option (See A2)

ALTERNATE FLOW

A1: Export to PDF

1. The System retrieves the selected information and transforms it to PDF format
2. The System prompts the User to download the PDF
3. The User can choose to download their now-exported PDF

ALTERNATE FLOW

A2: Export to Email

1. The System retrieves the selected information and transforms it to HTML format
2. The System emails the transformed information to the User's email address

TERMINATION

The User has either downloaded their PDF or the email has been sent.

POST CONDITION

System returns to dashboard with queries or filters still applied to the data.

REQUIREMENT 5: SEARCH

DESCRIPTION & PRIORITY

A user should be able to search across all data on the system – including processed articles, named entities, locations, and categories. This is a Priority Level 3 as it would be very useful in quickly obtaining important information for users.

USE CASE

SCOPE

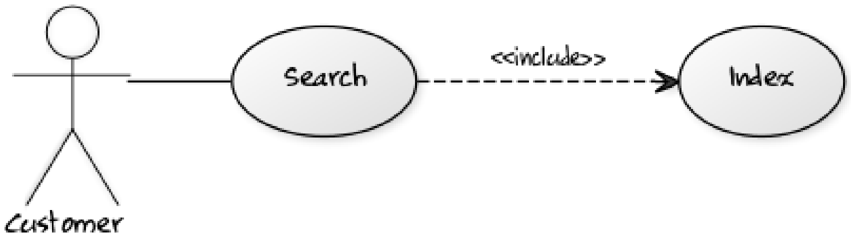
The scope of this use case that a user should be able to search across the entire Martello system from any page or dashboard area.

DESCRIPTION

A user should be able to perform a full-text search across the entire information stored in the Martello system – first and more important results are presented first while the rest of the search runs in the background.

USE CASE DIAGRAM

The following diagram outlines the actors and uses case involved in the Search requirement...



FLOW DESCRIPTION

PRECONDITION

The user must be logged into the system and have entered a global search field.

ACTIVATION

This use case starts when the User has typed in a search query and hit 'Enter'.

MAIN FLOW

1. User clicks into the Search box and types something
2. System runs a quick search of most recent information
3. System runs a search of indexed information from recent activity
4. System runs a full-text search across all articles for a certain time period

TERMINATION

The User has run the search and selected some information.

POST CONDITION

Search results are displayed, full-text background search has ceased.

NON-FUNCTIONAL REQUIREMENTS

The following non-functional requirements are expected of Martello…

PERFORMANCE/RESPONSE TIME REQUIREMENT

Martello will need to have a quick response time as it is a system that relies on the speed with which information is retrieved and presented to the end user. It is critical that articles are retrieved within 10 minutes of their posting to the relevant online media websites and processed by Martello for use by the end user.

AVAILABILITY REQUIREMENT

As with the performance and response time requirement, it is important that Martello is accessible at all times and has high-availability from an end-user perspective. Breaking news may need to be utilised in a business capacity at any point in the day or night, so Martello must not only have processed the information but also have it available to users at any point that they might require it.

MAINTAINABILITY REQUIREMENT

Martello will need to be easily maintainable from the perspective of administrators of the system, allowing them to troubleshoot any problems or issues with article retrieval, processing, or the frontend application that end users log onto.

EXTENDIBILITY REQUIREMENT

The system will have numerous aspects that will be of use to customers from launch; however, Natural Language Processing and Web Mining are fast moving technological areas, so it is important that Martello’s backend algorithms can be quickly altered and new ones implemented as the system evolves due to technological progress or customer feedback.

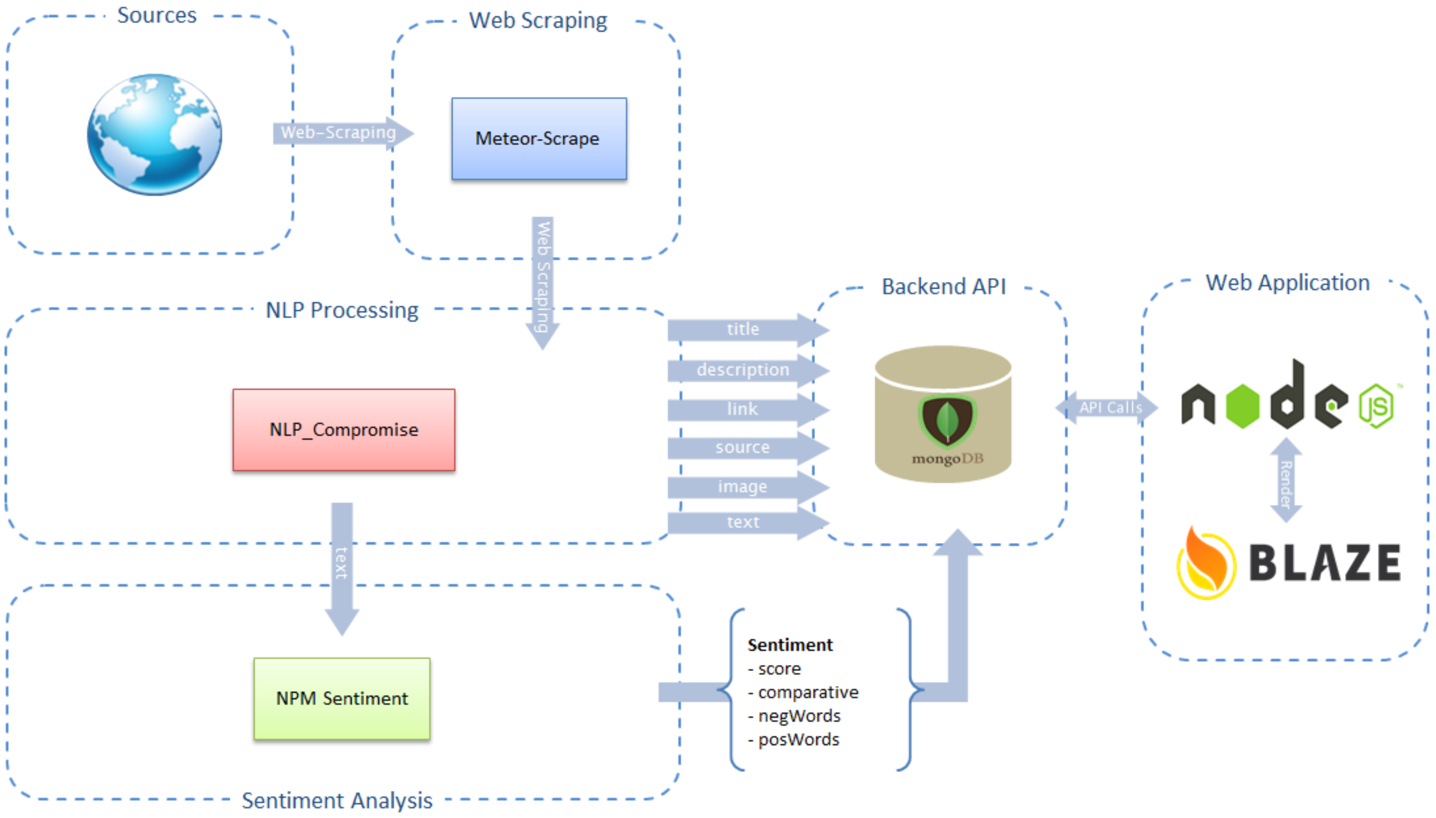
REUSABILITY REQUIREMENT

Information on the system will be stored across multiple databases, meaning that retrieval of information will always be easily available. The reuse of information, algorithms and other aspects (such as code or frontend widgets) will also be required in order to minimised system development overheads and to quickly advanced and produce new iterations of the software.

DESIGN AND ARCHITECTURE

ARCHITECTURE DIAGRAM

System Architecture diagram for the Martello system...



ARCHITECTURE OVERVIEW

Sources: for Martello's initial implementation, four sources are scraped for data – the websites for RTE, the Irish Times, the Irish Independent, and the Irish Examiner. These four sources were chosen as they represent the top four media outlets in Ireland and cover the widest range of news topics.

Web Scraping: the sources listed above are scraped every 15 minutes in order to ensure that the Martello database represents fresh articles and content. The web scraping itself is done through the use of *Meteor-Scrape* Node.js package.

NLP Processing: the Node.js package *NLP_Compromise* is used for Martello's natural language processing functions. This package allows the raw data scraped from sources to be parsed and turned into meaningful information. This information is passed to a MongoDB collection for access from the web application.

Sentiment Analysis: the Node.js package Sentiment is used for Martello's sentiment analysis function. The sentiment analysis function takes as its input the body text of each article scraped and passed through NLP Processing. This allows additional meaningful information to be extracted from the article text. Once it is processed by the sentiment analysis function this new information is passed to the same record in the MongoDB collection as the information extracted in the NLP phase.

Backend API: The processed data is then stored in a MongoDB collection where it is ready for use in the web application. The MongoDB collection can be accessed through the use of the Mongo APIs, which call JSON format data to be parsed.

Web Application: The Martello logic is created using Node.js, with frontend rendering being done through the use of Blaze.js. The Bootstrap CSS/JS framework is also used for the frontend.

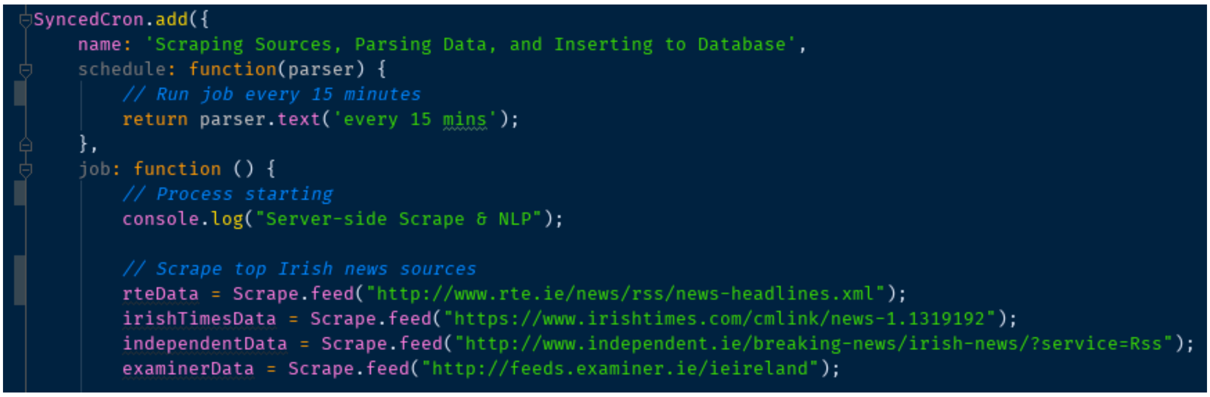
IMPLEMENTATION

SCRAPING

Meteor-Scrape is used for scraping the four principle sources that Martello uses. While there are currently four sources used this can easily be increased with a single line being added to the scrape function at the backend.

On top of this, the Node.js package *Synced-Chron* is used to schedule and run the source scrape. The schedule is set to every 15 minutes to ensure that fresh articles and content are continually being pulled into the system.

The following screenshot shows the Synced-Chron setup, followed by the scraping processes occurring…

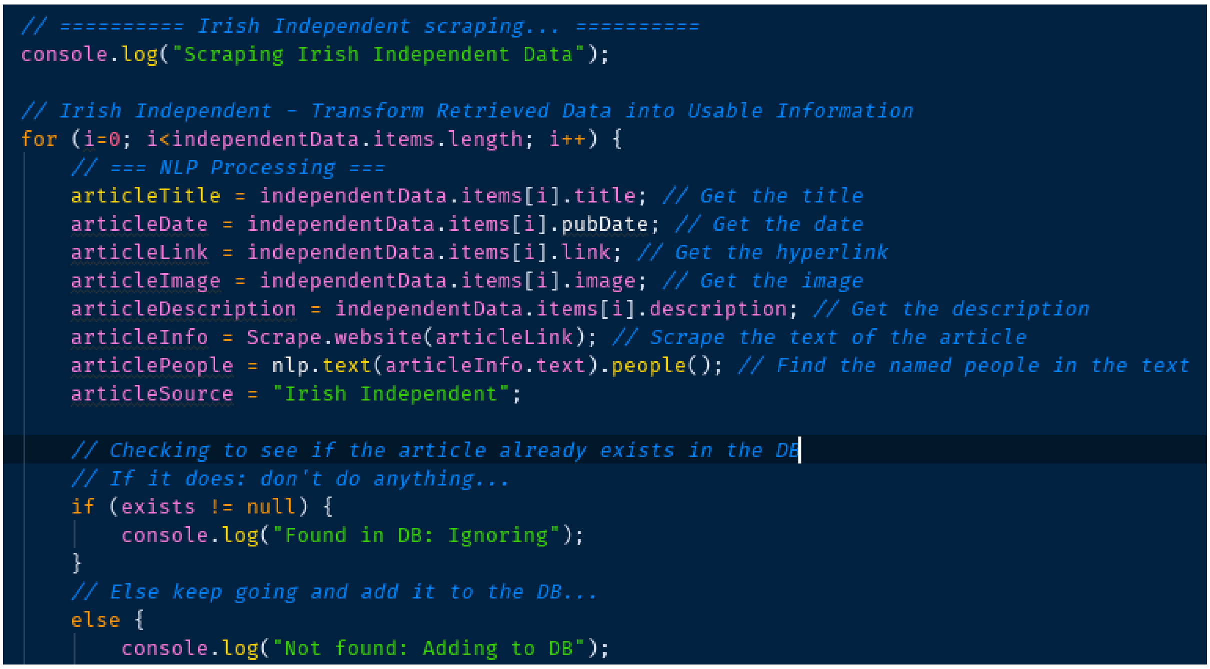


```
SyncedCron.add({
    name: 'Scraping Sources, Parsing Data, and Inserting to Database',
    schedule: function(parser) {
        // Run job every 15 minutes
        return parser.text('every 15 mins');
    },
    job: function () {
        // Process starting
        console.log("Server-side Scrape & NLP");

        // Scrape top Irish news sources
        rteData = Scrape.feed("http://www.rte.ie/news/rss/news-headlines.xml");
        irishTimesData = Scrape.feed("https://www.irishtimes.com/cmlink/news-1.1319192");
        independentData = Scrape.feed("http://www.independent.ie/breaking-news/irish-news/?service=Rss");
        examinerData = Scrape.feed("http://feeds.examiner.ie/ieireland");
```

NATURAL LANGUAGE PROCESSING

Once the information is scraped from the sources is it parsed using the NLP-Compromise Node.js package. This allows Martello to turn the raw retrieved data into meaningful information and forms a central component of the entire system.

The following screenshot shows retrieved data from the Irish Independent website being parsed and processed using NLP.

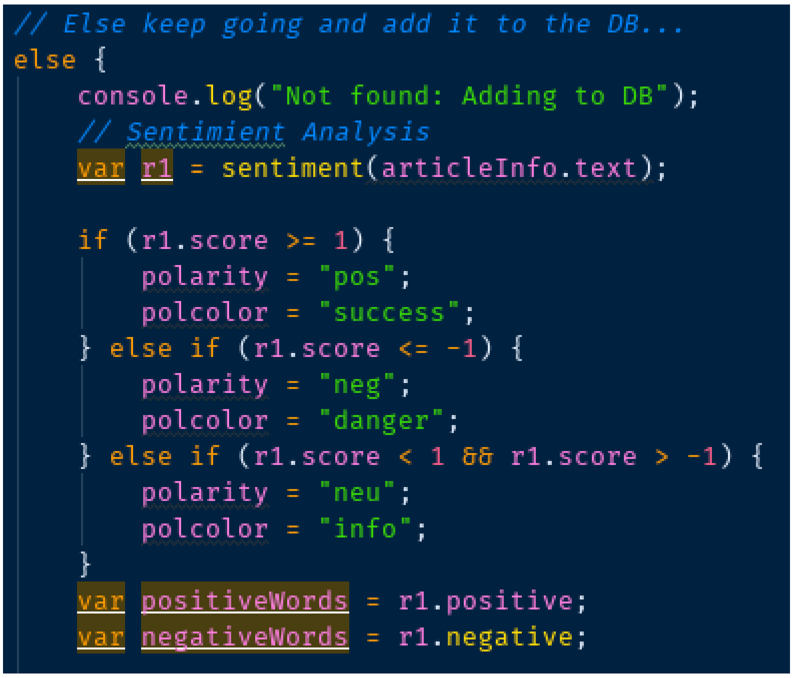


```
// ========== Irish Independent scraping... ==========
console.log("Scraping Irish Independent Data");

// Irish Independent - Transform Retrieved Data into Usable Information
for (i=0; i<independentData.items.length; i++) {
    // === NLP Processing ===
    articleTitle = independentData.items[i].title; // Get the title
    articleDate = independentData.items[i].pubDate; // Get the date
    articleLink = independentData.items[i].link; // Get the hyperlink
    articleImage = independentData.items[i].image; // Get the image
    articleDescription = independentData.items[i].description; // Get the description
    articleInfo = Scrape.website(articleLink); // Scrape the text of the article
    articlePeople = nlp.text(articleInfo.text).people(); // Find the named people in the text
    articleSource = "Irish Independent";

    // Checking to see if the article already exists in the DB
    // If it does: don't do anything...
    if (exists != null) {
        console.log("Found in DB: Ignoring");
    }
    // Else keep going and add it to the DB...
    else {
        console.log("Not found: Adding to DB");
```

After the information is extracted from the raw data a statement checks the current database to ascertain if it already contains the articles (so as to avoid duplication of content in the database). If the article is found the process ends for that article, but if it is new information it is passed on for sentiment analysis.

SENTIMENT ANALYSIS

Following the natural language processing component, the Node.js package Sentiment is used to analyse the sentiment of the retrieved article text. The following screenshot shows a part of this sentiment analysis, where articles with a score above 1 are given the polarity 'positive', articles with a score of lower than -1 are given a polarity of 'negative', and articles that fall in the band between the two are given a polarity of 'neutral'…



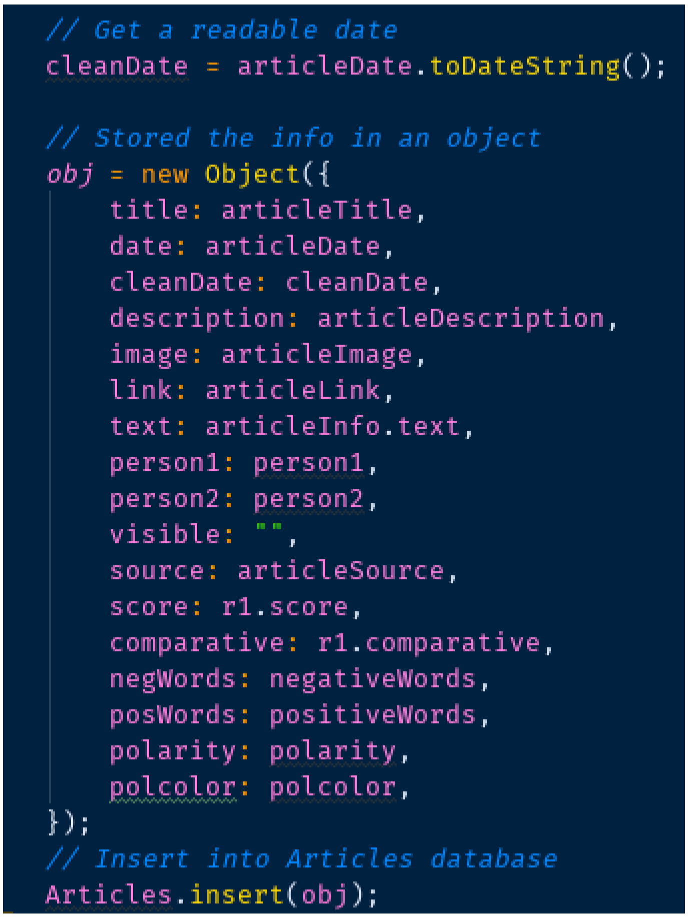
```
// Else keep going and add it to the DB...
else {
    console.log("Not found: Adding to DB");
    // Sentimient Analysis
    var r1 = sentiment(articleInfo.text);

    if (r1.score >= 1) {
        polarity = "pos";
        polcolor = "success";
    } else if (r1.score <= -1) {
        polarity = "neg";
        polcolor = "danger";
    } else if (r1.score < 1 && r1.score > -1) {
        polarity = "neu";
        polcolor = "info";
    }
    var positiveWords = r1.positive;
    var negativeWords = r1.negative;
```

The number of positive words and negative words are also extracted for use on the Martello frontend, this is used for frontend chart rendering.

DATABASE PERSISTENCE

Finally, having processed the raw data into meaningful information it is passed into a JavaScript object using key-value pairings.

This object is then inserted into the MongoDB Collection ‘Articles’ …



```
// Get a readable date
cleanDate = articleDate.toDateString();

// Stored the info in an object
obj = new Object({
    title: articleTitle,
    date: articleDate,
    cleanDate: cleanDate,
    description: articleDescription,
    image: articleImage,
    link: articleLink,
    text: articleInfo.text,
    person1: person1,
    person2: person2,
    visible: "",
    source: articleSource,
    score: r1.score,
    comparative: r1.comparative,
    negWords: negativeWords,
    posWords: positiveWords,
    polarity: polarity,
    polcolor: polcolor,
});
// Insert into Articles database
Articles.insert(obj);
```

GRAPHICAL USER INTERFACE (GUI)

The following sections present the graphical user interfaces that Martello uses.

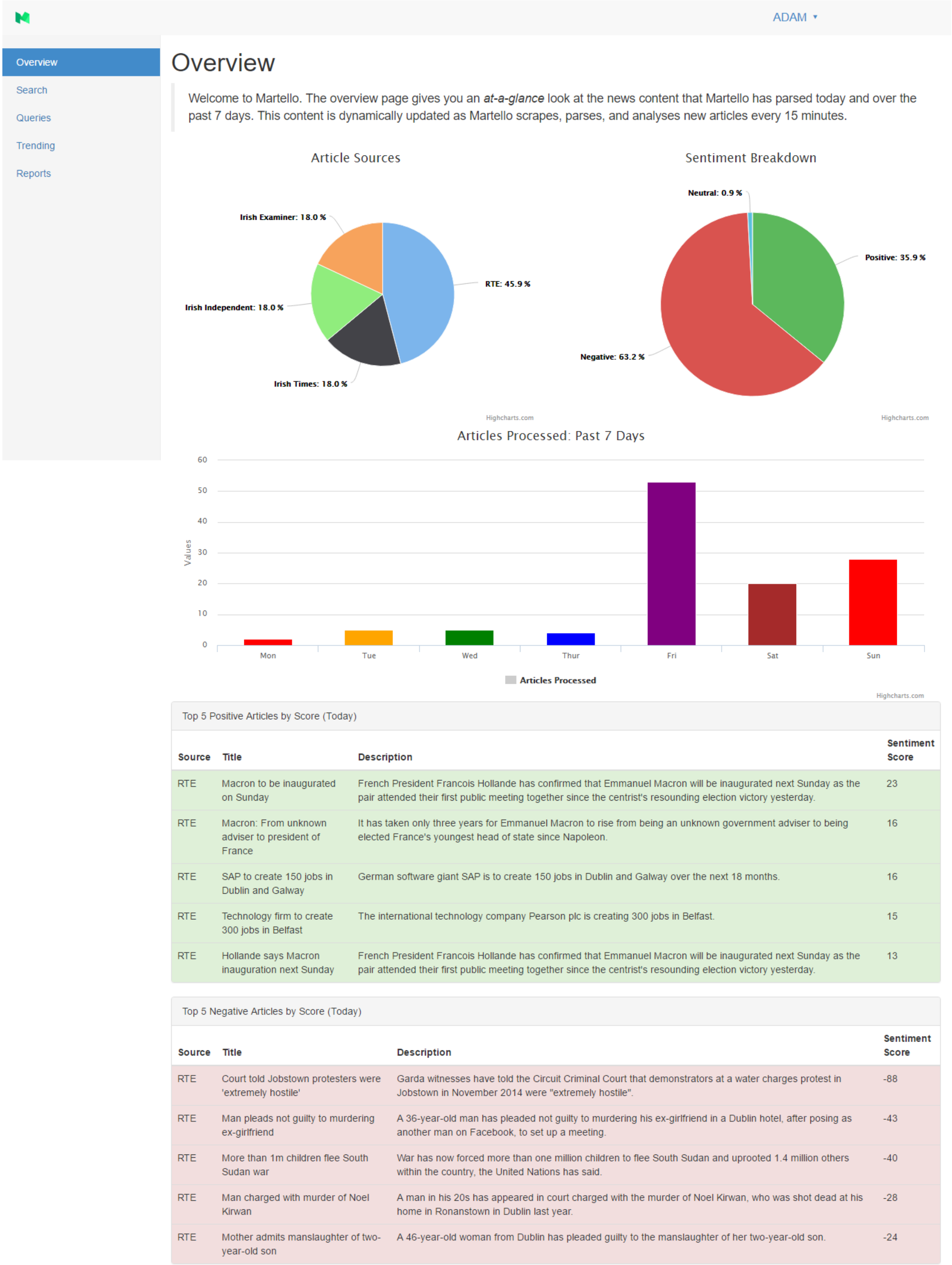
OVERVIEW

This screenshot presents the *Overview* section of Martello. The Overview page is dynamically generated from the current database state, and as such it reflects in real-time the articles that have been processed by Martello.

Included are charts that break down both the percentage of articles from each source as well as the overall sentiment value of the articles.

An additional bar chart displays the quantity of articles processed over the preceding 7 days. Again, all this data is dynamically fetched from the current MongoDB state.

Lastly, the Overview section shows both the 5 articles with the highest positive sentiment and the 5 articles with the lowest negative sentiment that day.



Overview

Welcome to Martello. The overview page gives you an *at-a-glance* look at the news content that Martello has parsed today and over the past 7 days. This content is dynamically updated as Martello scrapes, parses, and analyses new articles every 15 minutes.

Top 5 Positive Articles by Score (Today)

Source	Title	Description	Sentiment Score
RTE	Macron to be inaugurated on Sunday	French President Francois Hollande has confirmed that Emmanuel Macron will be inaugurated next Sunday as the pair attended their first public meeting together since the centrist's resounding election victory yesterday.	23
RTE	Macron: From unknown adviser to president of France	It has taken only three years for Emmanuel Macron to rise from being an unknown government adviser to being elected France's youngest head of state since Napoleon.	16
RTE	SAP to create 150 jobs in Dublin and Galway	German software giant SAP is to create 150 jobs in Dublin and Galway over the next 18 months.	16
RTE	Technology firm to create 300 jobs in Belfast	The international technology company Pearson plc is creating 300 jobs in Belfast.	15
RTE	Hollande says Macron inauguration next Sunday	French President Francois Hollande has confirmed that Emmanuel Macron will be inaugurated next Sunday as the pair attended their first public meeting together since the centrist's resounding election victory yesterday.	13

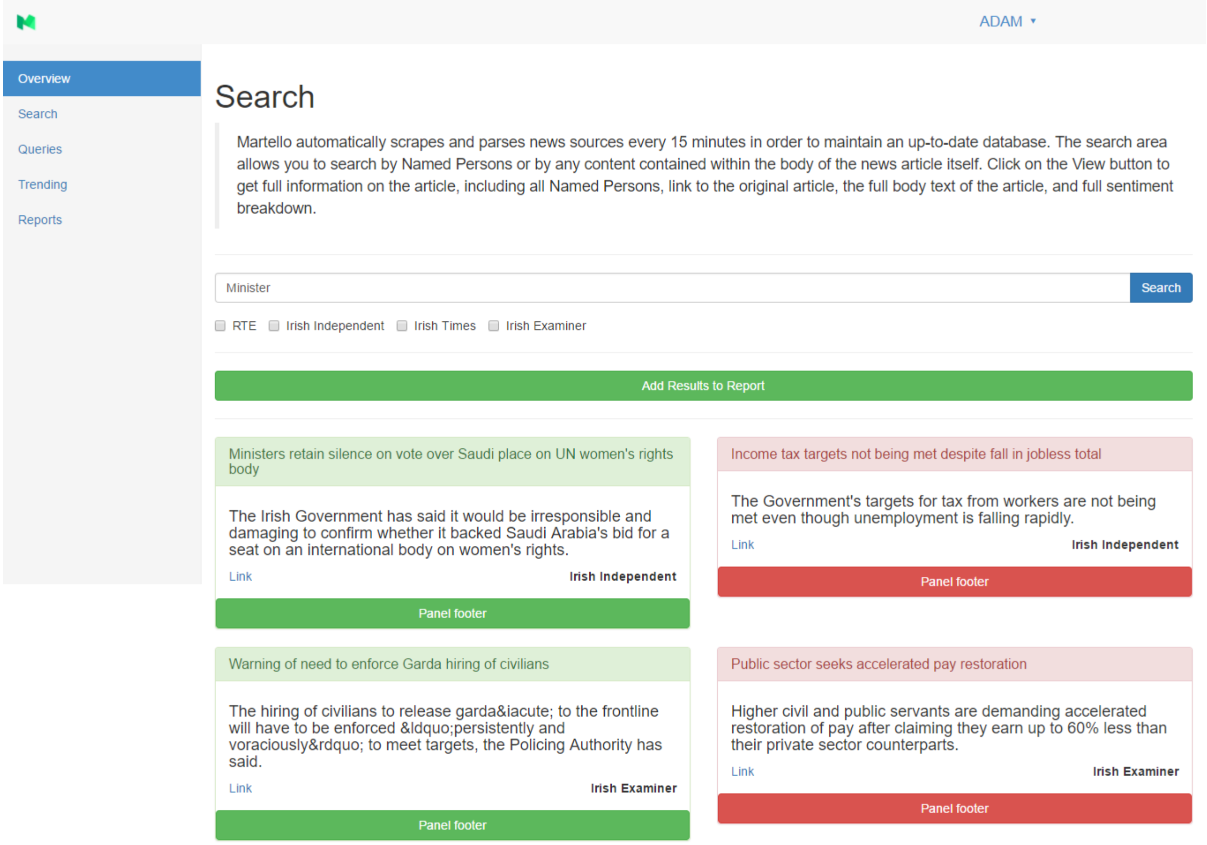
Top 5 Negative Articles by Score (Today)

Source	Title	Description	Sentiment Score
RTE	Court told Jobstown protesters were 'extremely hostile'	Garda witnesses have told the Circuit Criminal Court that demonstrators at a water charges protest in Jobstown in November 2014 were "extremely hostile".	-88
RTE	Man pleads not guilty to murdering ex-girlfriend	A 36-year-old man has pleaded not guilty to murdering his ex-girlfriend in a Dublin hotel, after posing as another man on Facebook, to set up a meeting.	-43
RTE	More than 1m children flee South Sudan war	War has now forced more than one million children to flee South Sudan and uprooted 1.4 million others within the country, the United Nations has said.	-40
RTE	Man charged with murder of Noel Kirwan	A man in his 20s has appeared in court charged with the murder of Noel Kirwan, who was shot dead at his home in Ronanstown in Dublin last year.	-28
RTE	Mother admits manslaughter of two-year-old son	A 46-year-old woman from Dublin has pleaded guilty to the manslaughter of her two-year-old son.	-24

SEARCH

SEARCH RESULTS

The Search page allows users to search the articles in the Martello database by either named persons (e.g. Enda Kenny, Donald Trump) or through any text contained in the body of the article. IN this screenshot the term “Minister” has been searched for (only partial search results are shown in this screenshot)…



Search

Martello automatically scrapes and parses news sources every 15 minutes in order to maintain an up-to-date database. The search area allows you to search by Named Persons or by any content contained within the body of the news article itself. Click on the View button to get full information on the article, including all Named Persons, link to the original article, the full body text of the article, and full sentiment breakdown.

Minister [Search]

□ RTE □ Irish Independent □ Irish Times □ Irish Examiner

Add Results to Report

Ministers retain silence on vote over Saudi place on UN women's rights body

The Irish Government has said it would be irresponsible and damaging to confirm whether it backed Saudi Arabia's bid for a seat on an international body on women's rights.

Link — Irish Independent

Panel footer

Income tax targets not being met despite fall in jobless total

The Government's targets for tax from workers are not being met even though unemployment is falling rapidly.

Link — Irish Independent

Panel footer

Warning of need to enforce Garda hiring of civilians

The hiring of civilians to release garda&iacute; to the frontline will have to be enforced &ldquo;persistently and voraciously&rdquo; to meet targets, the Policing Authority has said.

Link — Irish Examiner

Panel footer

Public sector seeks accelerated pay restoration

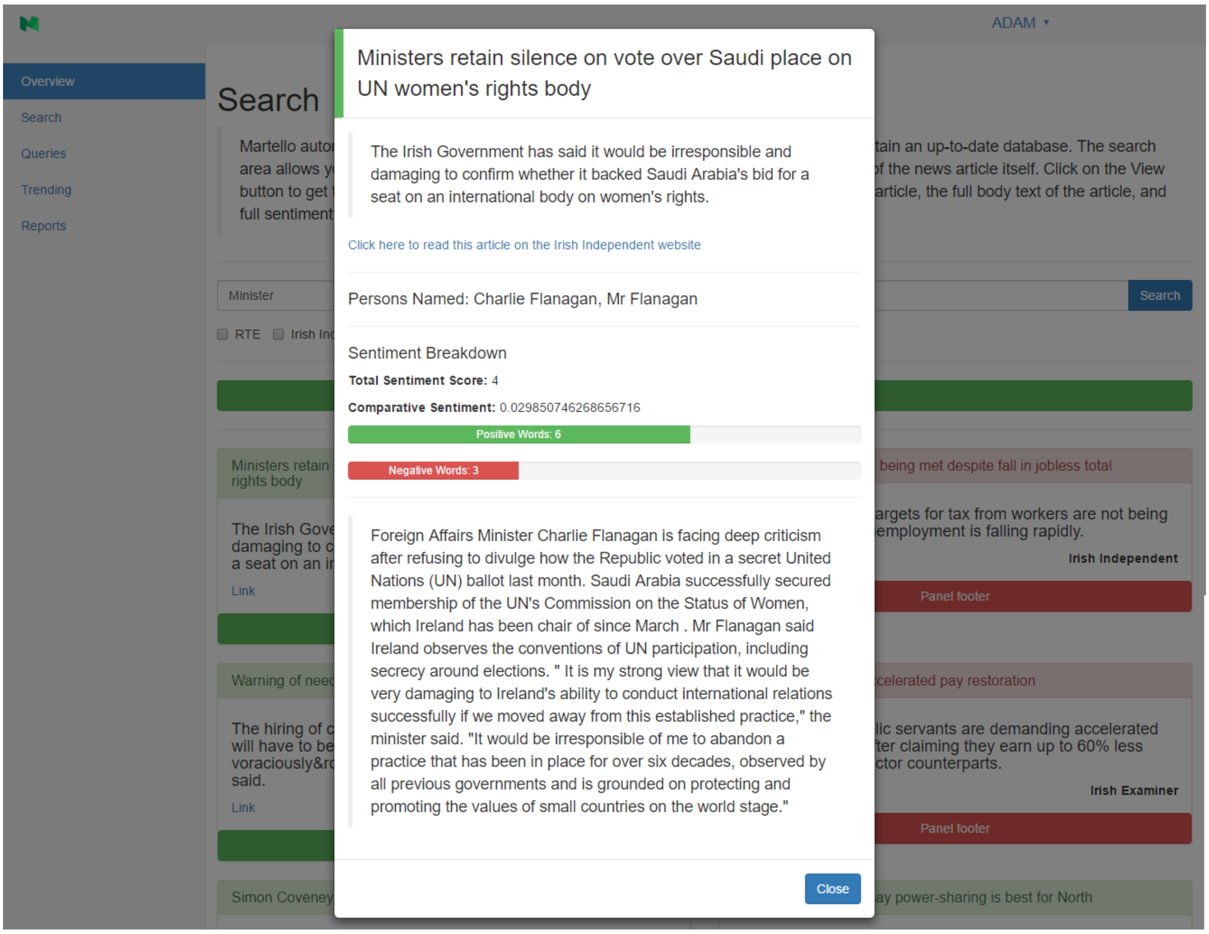
Higher civil and public servants are demanding accelerated restoration of pay after claiming they earn up to 60% less than their private sector counterparts.

Link — Irish Examiner

Panel footer

INDIVIDUAL ARTICLE

When a user clicks on an article to view it a modal pops up displaying more in-depth information about that article, including all named persons, the full text of the article, a link to the original article, and a number of sentiment analytics…



Ministers retain silence on vote over Saudi place on UN women's rights body

> The Irish Government has said it would be irresponsible and damaging to confirm whether it backed Saudi Arabia's bid for a seat on an international body on women's rights.

Click here to read this article on the Irish Independent website

Persons Named: Charlie Flanagan, Mr Flanagan

Sentiment Breakdown

Total Sentiment Score: 4

Comparative Sentiment: 0.029850746268656716

Positive Words: 6

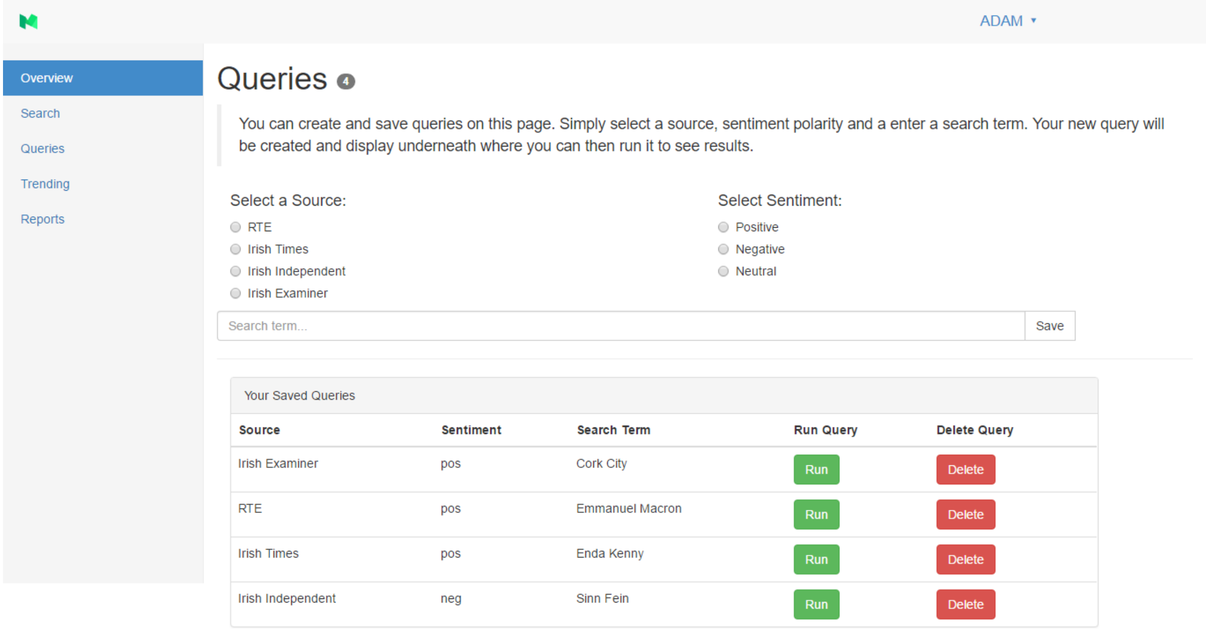
Negative Words: 3

> Foreign Affairs Minister Charlie Flanagan is facing deep criticism after refusing to divulge how the Republic voted in a secret United Nations (UN) ballot last month. Saudi Arabia successfully secured membership of the UN's Commission on the Status of Women, which Ireland has been chair of since March . Mr Flanagan said Ireland observes the conventions of UN participation, including secrecy around elections. " It is my strong view that it would be very damaging to Ireland's ability to conduct international relations successfully if we moved away from this established practice," the minister said. "It would be irresponsible of me to abandon a practice that has been in place for over six decades, observed by all previous governments and is grounded on protecting and promoting the values of small countries on the world stage."

Close

QUERIES

The Queries section allows users to create quick queries that can be run at the click of a button. Users can enter a search term, select a source and a sentiment polarity, and then save that query to their account. When they wish to run the query they just log back in an select Run on their specific query…



Queries 4

You can create and save queries on this page. Simply select a source, sentiment polarity and a enter a search term. Your new query will be created and display underneath where you can then run it to see results.

Select a Source:

○ RTE
○ Irish Times
○ Irish Independent
○ Irish Examiner

Select Sentiment:

○ Positive
○ Negative
○ Neutral

Your Saved Queries

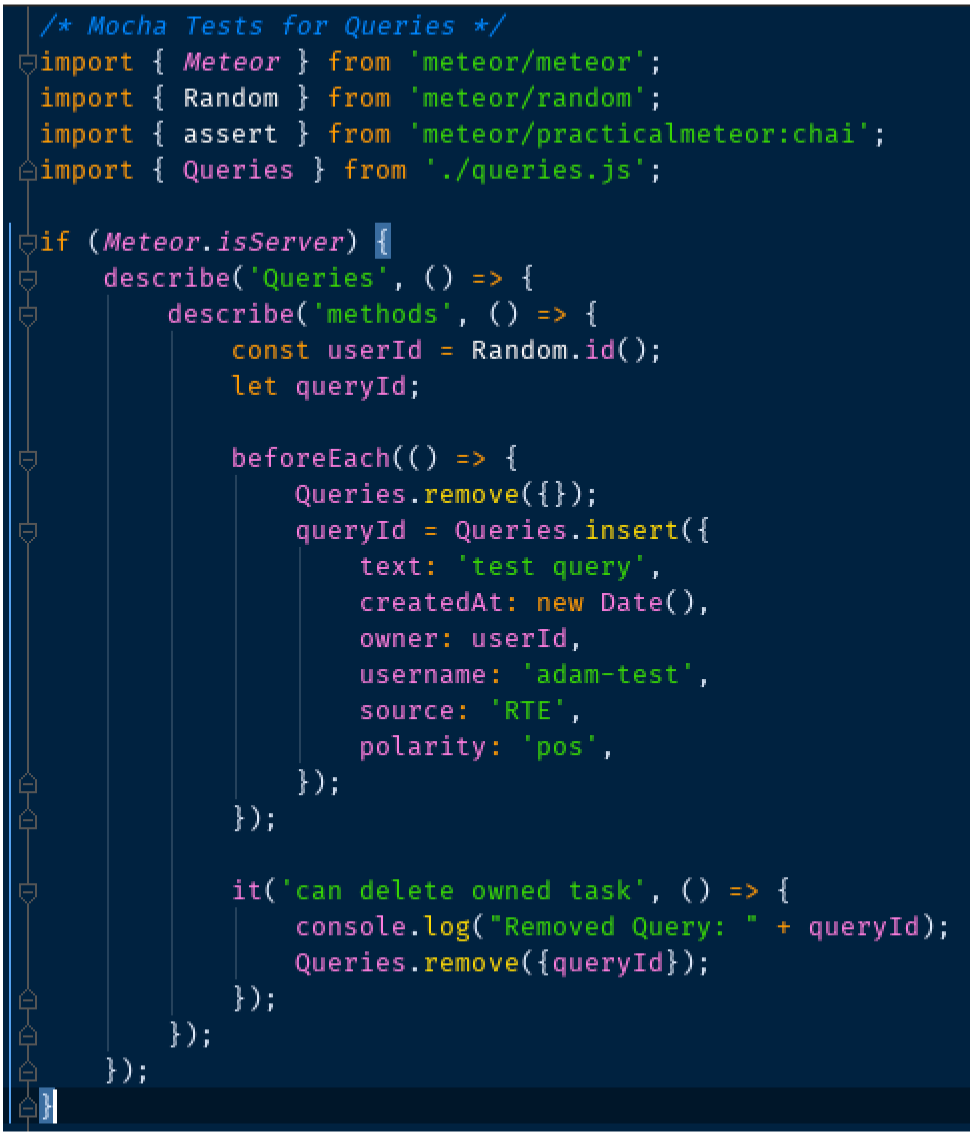
Source	Sentiment	Search Term	Run Query	Delete Query
Irish Examiner	pos	Cork City	Run	Delete
RTE	pos	Emmanuel Macron	Run	Delete
Irish Times	pos	Enda Kenny	Run	Delete
Irish Independent	neg	Sinn Fein	Run	Delete

TESTING

Martello was developed with using a Test-Driven Development philosophy. As such, it was important that unit test were run throughout the entire development process. This testing was implemented and carried out using the robust unit testing pairings of the test runner Mocha.js and the assertion library Chai.js.

UNIT TESTS

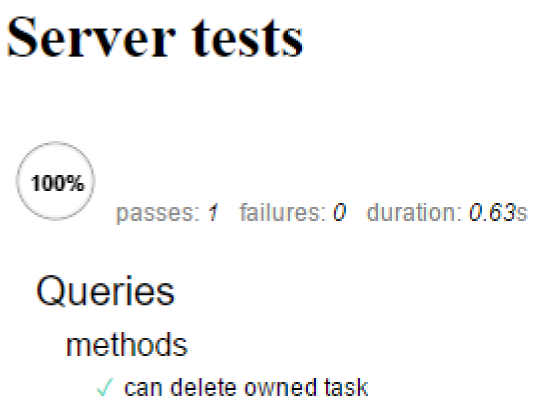
This screenshot displays a Mocha.js unit test that is testing to ensure that the Queries database collection handles document insertion and removals (this is a single test, further tests are carried out for each MongoDB collection and also for client-side Blaze template rendering)…



```
/* Mocha Tests for Queries */
import { Meteor } from 'meteor/meteor';
import { Random } from 'meteor/random';
import { assert } from 'meteor/practicalmeteor:chai';
import { Queries } from './queries.js';

if (Meteor.isServer) {
    describe('Queries', () => {
        describe('methods', () => {
            const userId = Random.id();
            let queryId;

            beforeEach(() => {
                Queries.remove({});
                queryId = Queries.insert({
                    text: 'test query',
                    createdAt: new Date(),
                    owner: userId,
                    username: 'adam-test',
                    source: 'RTE',
                    polarity: 'pos',
                });
            });

            it('can delete owned task', () => {
                console.log("Removed Query: " + queryId);
                Queries.remove({queryId});
            });
        });
    });
}
```

TEST RESULTS

And using the mocha test-driver we can load up our testing environment on localhosts to see the results of this database test…



Server tests

100%

passes: 1 failures: 0 duration: 0.63s

Queries

methods

✓ can delete owned task

This server-side unit testing is run for all MongoDB collections (Queries, Articles, Reports, etc). Additional frontend tests are run for to ensure Blaze templates are rendering correctly on the client.

FURTHER DEVELOPMENT

There are many ways in which Martello may be of use to journalists and other media professionals in its initial format; however there is always scope for improvement, updates, and evolution of the system. The following areas are ones in which it is felt that Martello will be able to advance its functions and make improvements in in the future.

NEW NLP ALGORITHMS:

While Martello will be looking at a number of Natural Language Processing tasks at first – including Sentiment Analysis, Named Entity Recognition, Categorisation and Summarisation – there are many other NLP algorithms available, with the number growing all the time.

Some of these include Discourse Analysis, a method by which NLP algorithms can deduce the discourse structure of a corpus of text discovering how sentences relate to one another and are used to make and reinforce points, Machine Translation, and Relationship Extraction – which is a method used to discover relationships between named entities in a text that aren't explicitly stated or defined.

With this in mind there is scope for Martello to add additional NLP processing tasks to its suite of functions in future, increasing its power, customer reach, and even industries of use.

NATURAL LANGUAGE GENERATION

In its first iteration Martello will retrieve information from many online news and media outlets, processes them, and gives journalists and others a brief overview and synopsis of that information for use in their own organisations. However, it is envisioned that Martello will build on this platform to become a system that does much more than simple retrieval and interpretation.

The area of Natural Language Generation is a growing one, and with the advent of personal AI assistants, chatbots, and other computer-based systems that can not only understand, but also *generate* their own natural language, it is hoped that in future Martello will be able to move into this area also.

Natural Language Generation through Machine Learning is an area that needs vast quantities of written and tagged data in order to generate human language. Martello's initial suite of offerings is essentially one massive process that undertakes the initial collection of the information required to later use Machine Learning algorithms for Natural Language Generation.

As such, it is proposed that at some stage this collected and tagged data will be put to use in the form of allowing ML algorithms to use it to produce its own written articles based on trending news topics. Whereas Martello in its first iteration will allow journalists to produce their own articles much easier and more quickly than before, it is hoped that down the line Martello will have already written most of their article for them, leaving the journalist to simply add their own stamp or 'flavour' to their news content.

OTHER SOCIAL MEDIA ANALYSES

While Martello at first will focus on analysis tweets from Twitter as an avenue to see what the prevailing sentiment is across social media, we are aware that there are other social media networks that may hold valuable information that Martello may make use of in order to better gauge the mood with regards to certain topics.

With this in mind, it is proposed to add Facebook and other social media outlets to Martello’s range of consumed news sources. This will allow Martello to get a much wider-ranging set of information to analyse and incorporate into its system.

CONCLUSIONS

Ultimately, Martello will be a difficult software system to implement due to many factors. Firstly, from an architectural standpoint, it will really be comprised of two distinct systems connected via an API – the pre-processing system and the frontend web application itself.

The pre-processing system, which takes cares of web-scraping and natural language processing algorithms, will be challenging as it will mean dealing with non-trivial technologies and incorporating them into a robust software system. The frontend will be consumer facing and as such will require a well thought out user experience and interface, which is quick to respond to user input and queries.

While have two systems in this manner may add some work in the engineering end of things, the separation of technical concerns will future-proof Martello in the case of the needing to either switch natural language processing libraries or to rewriting the frontend web application for design and functionality purposes. However, despite the relative technological complexity, I feel that, once completed, Martello will be a Software as a Service product that undoubtedly has commercial viability.

Many current companies and start-ups are using natural language processing algorithms for innovative and interesting purposes, but I feel that Martello's focus on divining meaning from text corpora for media and press professionals is a newer slant on this, with many current offers focusing on sales lead generation and advertising.

Overall, Martello will be difficult to develop and deploy, but the technologies involved, along with the scope and idea itself, gives me confidence that it will be a success.

REFERENCES

Bird, S., & Loper, E. (2004). NLTK: The Natural Language Toolkit. *Proceedings of the ACL demonstration session*, 214-217.

Bird, S., Klein, E., & Lope, E. (2009). *Natural language processing with Python.* O'Reilly Media, Inc.

Kane, F. (2016, September). Data Science and Machine Learning with Python. Udemy.

Kiser, M. (2016, August 11). *Introduction to Natural Language Processing (NLP) 2016*. Retrieved September 26, 2016, from algorithmia.com: http://blog.algorithmia.com/introduction-natural-language-processing-nlp/

Kumar, V. (2016, October). MEAN Stack For Web Developers. Udemy.

Madnani, N. (2007). Getting Started on Natural Language Processing with Python. *ACM Crossroads, 13*(4).

Meade, A. (2016, October). The Complete Node.js Developer Course 2.0. Udemy.

Sacash, B. (2016, October). Introduction to Natural Language Processing. Udemy.

APPENDIX

PROJECT PROPOSAL

Final Year Project Proposal

Web application utilising Web Mining and Natural Language Processing techniques to monitor, analyse, and interpret news articles and social media posts for journalists and other media professionals

BACKGROUND

As the quantity of news content, articles and other media produced each day online grows, press officers, journalists and other media professionals across various sectors find it increasing difficult to stay abreast of news critical to their roles. Many rely on manually combing through relevant news sources each morning, while others use third-party companies which essentially outsources this for them – neither solution being optimal.

Complicating matters is the nature of the 24 hour news cycle, whereby it's difficult for individuals to keep up with the sheer quantity of news being produced on a daily basis, and the various social media platforms which essentially produce constant streams of new content, opinions and consumer sentiment.

These factors combine to create a fast-moving and constantly-changing media environment for journalists, press officers and brand representatives, with the current state of things often being elusive and hard to nail down.

IDEA

With the increase in processing power and constantly improving algorithms, computers are getting progressively better at processing unstructured text, with many libraries across multiple programming languages now able to parse text, social media posts, html pages, and various other sources of natural language to infer meaning, sentiment and many other attributes previously only possible by human means.

Through Natural Language Processing (NLP) libraries it is possible for computers to carry out much of the preliminary work that once fell to journalists and other media professionals. Analysing news stories across multiple media sources in real time can be done much faster by NLP algorithms than by individual journalists.

With the above in mind, I am proposing to develop a cloud-based web application that monitors and scrapes news articles and posts from the top news and social media outlets online, and then processes these articles using Python's Natural Language Toolkit (NLTK) in order to determine:

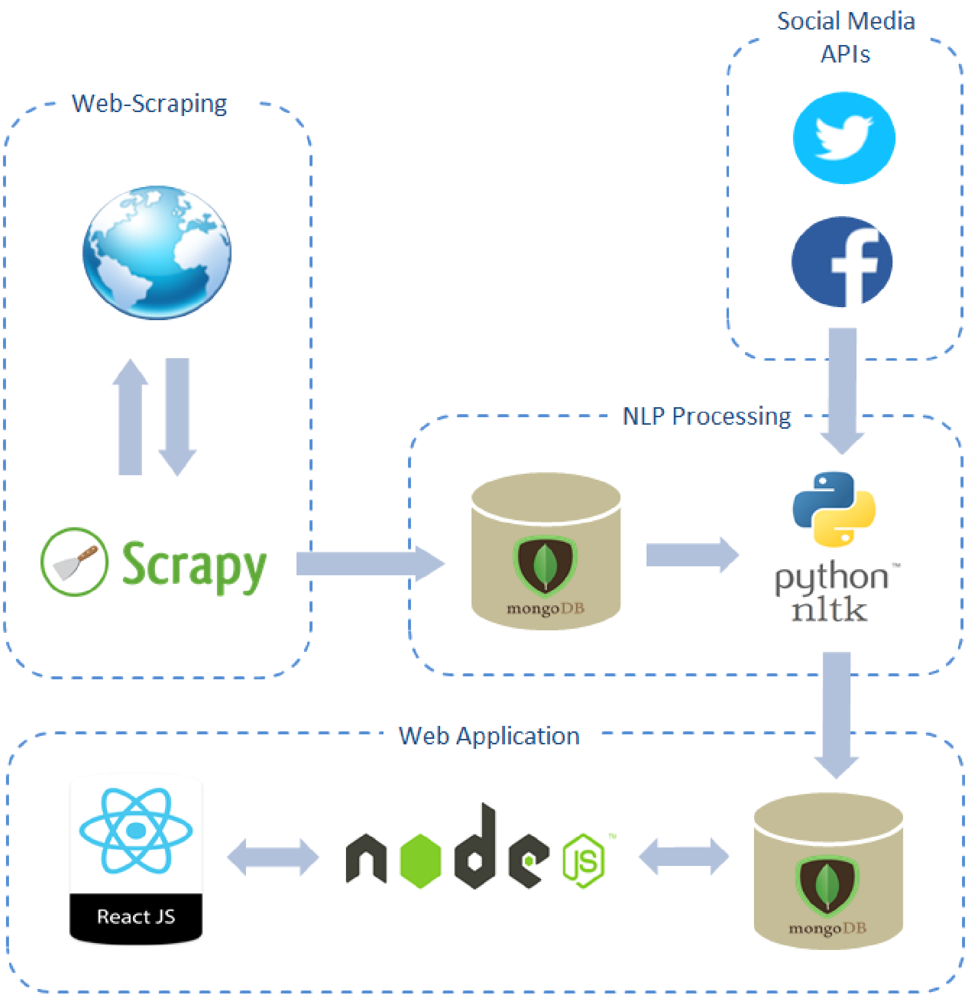
- Named Entities (e.g. Enda Kenny, National College of Ireland, Microsoft, etc)
- Sentiment (positive or negative intent behind the text)
- Categories
- Summaries

The processed, and now structured, information will then be deposited into a database that is accessible from a web application frontend by the end user.

TECHNOLOGIES & APPLICATION

POTENTIAL TECHNOLOGY STACK AND ARCHITECTURE

The following diagram illustrates a potential technology stack and architecture that could be used for this project.



WEB-SCRAPING

A directory of online media and news outlets will be compiled in order to find the most relevant and influential in each locale. Outlets that expose their data publicly via an API will be queried for the information required at the Natural Language Processing stage. As many sites will not have APIs implemented the news or RSS feeds of these websites will be scraped using Scrapy, a Python web-scraping library.

SOCIAL MEDIA APIS

The APIs of the main social media platforms will be used to obtain posts, articles, microposts and comments which will then be further processed using NLP. This aspect of the application will complement the previous scraping and analysing of news articles, in that it will identify the level of social engagement of topics and how they are being perceived by the general public (as opposed to professional journalists).

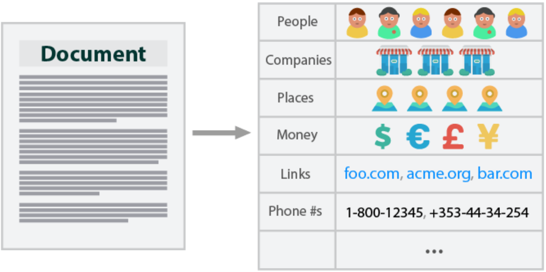
NLP PROCESSING

The retrieved news and media outlet information will then be stored in a MongoDB database before it is processed using Python's Natural Language Toolkit (NLTK). Social media content will be passed directly through NLTK and into the web-application database.

NLTK will process the content for various attributes and meaning. It is also worth noting that the below listed attributes are a small subset of the many possible factors that NLTK can identify in a text corpus, and that many others may be of use or interest in this application.

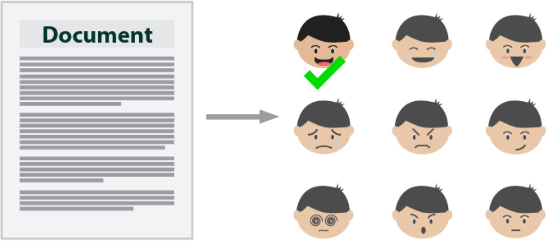
NAMED ENTITY RECOGNITION

Named Entity Recognition identifies proper names from the scraped information and will allow the end user to filter by specific people, organisations and locations.



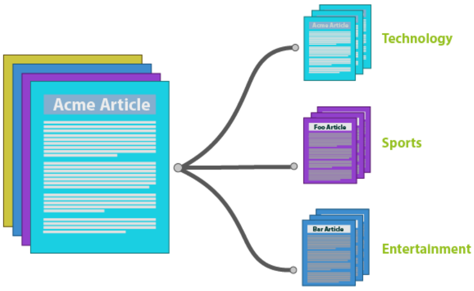
SENTIMENT ANALYSIS

Sentiment analysis gives a rating as to level of positive or negative sentiment behind a piece of text, allowing users to quickly observe the prevailing thought about certain topics, subjects or people.



CATEGORISATION

Categorisation identifies pieces of text as falling into predefined categories, allowing the end user to search according to specific tags and categories in order to narrow down the volume of information.



SUMMARISATION

Summarisation condenses a text down to its most important sentences and information, allowing the end user to quickly comprehend the substance of an article without having to read through the entire content.

WEB APPLICATION

NLTK will pass the processed information (now in a structured format) into a MongoDB database which is directly connected to the web application. The processed social media information will also be stored in a MongoDB database. The application itself will be built using NodeJS for the server and BlazeJS for the frontend (on top of the ExpressJS framework).

RESOURCES REQUIRED

The programming languages and technologies used to build Martello.io are largely free and open-source, with the only expenses coming from web and database hosting, and domain registration. The following resources will be required throughout the project.

Resource Type	Resources
Programming Languages:	Python and JavaScript
Frontend:	HTML5 and CSS3
Backend:	NodeJS and BlazeJS
Pre-Processing:	Scrapy and Natural Language Toolkit (NTLK)
Web App Hosting:	Heroku
Database Hosting:	MongoLab
Domain Registration:	GoDaddy

MARKET

As this application will be built using some of the latest technologies it will be used for a number of novel and interesting tasks. Some of these are listed below, but it is envisioned that as the application matures and evolves it will be utilised for purposes well beyond the scope of this document.

POTENTIAL USE CASES

PRESS OFFICE

In many organisations the Press Office needs to stay informed of any news articles pertaining to specific individuals. Using the web application proposed in this document, staff members will be able to save the names of these people as keywords in the system, quickly able to retrieve and view all the latest news articles about that person.

An example of this could be that the Press Office at the Department of Transport might want to save keyphrases such as "Shane Ross", "Minister for Transport", "Dublin Bus Strike", etc, in order to quickly find news stories about those keyphrases. Staff members could then look over the returned article summaries, as well as the sentiment analysis, in order to see the prevailing mood towards each in the media.

JOURNALIST

While Press Offices may use the application for observation and reaction, journalists themselves could also use it for research and sourcing of new article subjects and topics.

For example, a journalist at a smaller media outlet would not have the on-the-ground resources of Reuters, the Associated Press or the BBC in the case of breaking news stories from conflict zones. Rather than having to search through various – often conflicting – news sources to ascertain what is happening, the journalist could use this application to quickly source information by viewing summaries of the critical information in each article, in turn allowing them to have a quicker turnaround for their own story.

COMPETITORS

NEWSWHIP

Newswhip are an Irish company focusing on online content discovery and social engagement of news stories. While Newswhip firmly exists in the news and social media space they seem to operate predominantly on discovery of stories in any area that have the potential for high social velocity and virality online, as opposed to the monitoring of all stories pertaining to specific companies, industries or people.

AYLIEN

Aylien is another Irish-based company that is a provider for Natural Language Processing APIs. While operating as a vendor for NLP and Machine Learning algorithms and tools, Aylien itself does not have its own web application harnessing and utilising these algorithms, and instead focuses on selling these tools to developers in order for them to create their own applications.

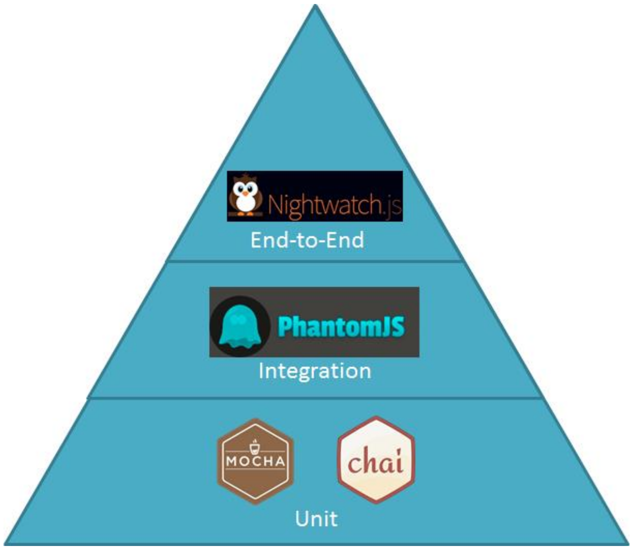
KANTAR MEDIA

Kantar Media are an international company with a presence in a number of markets. Kantar's *Media Monitoring* solution most closely aligns with the functionality of the proposed application; however, Kantar also incorporate a human element into their solution which increases costs and adds additional time to the analysis, both of which Martello.io is intended to minimise when in production.

EVALUATION

TESTING

Testing will be carried out according to the test pyramid, whereby the majority of tests are unit tests, followed by integration tests, and finally end-to-end tests.



UNIT TESTING

Unit Testing will be employed within the application codebase to test exposed methods on the backend. One of the most robust unit testing pairings for Node.js applications is the use of the test runner Mocha.js and the assertion library Chai.js.

INTEGRATION TESTING

Integration Testing will be used to test the software components in order to ensure that they function correctly prior to deployment. The headless, scriptable browser Phantom.js will be used here to run integration tests.

END-TO-END TESTING

End-to-End Testing will be used to test the entire application as it would function to an end user. The testing library Nightwatch.js will be used here to manipulate DOM elements as well as run functional tests on frontend components and input fields.

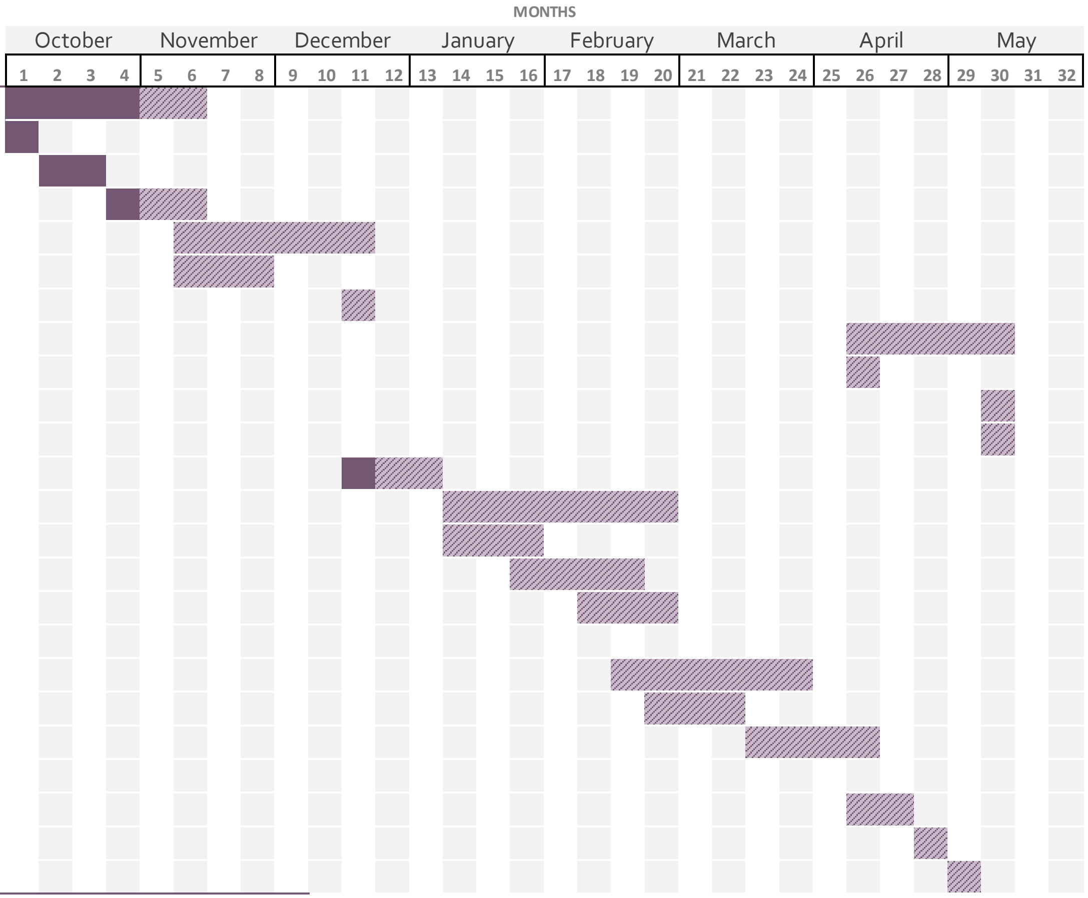
DATA APPRAISAL

There is a certain subjectivity to the information that is ultimately outputted from the system. As such, queries will have to be run against various types of input parameters in order to appraise the quality of the output (and, by extension, underlying data) that the system returns when being used by end users. Every aspect of the system that a user can specify filters or statements for will be manually tested to ensure that the output is clear, coherent and accurately reflects the nature of the question posed/query entered.

PROJECT PLAN

The following Gantt chart outlines the Project Plan including starting points, estimated durations, and key stages...

ACTIVITY	PLAN START	PLAN DURATION	PERCENT COMPLETE	
Initial Documentation	1	6	70%	🔴
Project Pitch	1	1	100%	🟢
Project Proposal	2	2	100%	🟢
Requirements Spec	4	3	10%	🔴
Mid-Point	6	6	0%	🔴
Prototype	6	3	0%	🔴
Mid-Point Presentation	11	1	0%	🔴
Final Submissions	26	5	0%	🔴
Showcase Materials	26	1	0%	🔴
Final Project	30	1	0%	🔴
Final Presentation	30	1	0%	🔴
Technology Stack Research	11	3	25%	🔴
Pre-Processing	14	7	0%	🔴
Web Scraping	14	3	0%	🔴
Natrual Language Processing	16	4	0%	🔴
Social Media APIs	18	3	0%	🔴
Web Application	0	3	0%	🔴
Node.js Backend	19	6	0%	🔴
MongoDB Connection	20	3	0%	🔴
React.js Frontend	23	4	0%	🔴
Deployment	0	3	0%	🔴
Testing	26	2	0%	🔴
Evaluation	28	1	0%	🔴
Cloud Deployment	29	1	0%	🔴



MONTHLY JOURNALS

SEPTEMBER 2016

Adam O’Callaghan
BSc in Computing (Evening) – 4th Year

MY ACHIEVEMENTS

This was the first month of the Software Project. The idea that I proposed and had ratified was one that I have had in mind in one form or another for a couple of years. I am feeling exciting and enthusiastic to get started on it.

The project itself involves a number of cutting-edge technologies, such as Natural Language Processing, Web Mining, and potentially Machine Learning incorporating Neural Networks – all technologies with which I hope to become sufficient in, and maybe even further explore to a Master’s Degree level.

MY REFLECTION

Creating a document and slides for the Project Ratification clarified some concepts in my head, while solidifying – and in some cases dismissing – some others. This will hopefully bode well down the line, as previously some of my ideas may have been underdeveloped. It’s good to see my ideas down on paper as previously they had simply been percolating in my head where I was unable to critically evaluate them.

I feel that I have a great idea on my hands, but that the execution will be difficult and challenging – but, hopefully, above all rewarding.

INTENDED CHANGES

I need to flesh out various aspects of my project – for example, while I have a “50,000 feet” overview of my technology stack and architecture I do need to explore these further.

I also need to read more about the individual technologies in order to better understand them and see what alternatives are out there, since in some cases there may be a better fit for certain aspects of the project than I had previously envisioned.

Time management is another area in which I need to excel, this semester is stacked with continuous assessments and, along with work and other commitments, the entire class will be very busy and need to use our time optimally.

OCTOBER 2016

Student name: Adam O'Callaghan
Programme: BSc in Computing (Evening) – 4th Year

MY ACHIEVEMENTS

This month I finished off my project proposal and submitted it. I feel that it was well formed and hopefully gives a much better indication of the level of project that I'm hoping to produce. It also helps that I have the ball rolling on the documentation now, as an entirely blank slate can be very daunting.

I stated last month that I wanted to delve deeper into the technologies I'm hoping to use to build my application – I have managed to do this, taking a number of Udemy online courses in NodeJS and BlazeJS, which were very informative and will help me when it comes time to build my prototype before the end of the semester.

MY REFLECTION

I'm going to continue to finish my documentation to a high level – not only with an eye to getting a good mark, but also because it becomes much clearer as you develop documentation how it will help you down the line. It can be frustrating sometimes to do proposals, architecture diagrams and requirements, but once they're completed it really does give a much better idea of where you're going with things.

I do need to get some consistency with studying, doing tutorials, and writing documentation. As it stands I tend to 'blitz' by doing a lot of work in a single period, rather than being consistent and doing smaller chunks each day. It leaves me under a lot of pressure, even though I feel the standard of my output is still good.

INTENDED CHANGES

Going forward I will ensure that I am consistent in studying for continuous assessments instead of 'blitzing' it in massive chunks. I'll be continuing to do the Udemy and Pluralsight tutorials, and getting started on my prototype. I have already started my Requirements Specification so I will finish this off and get it uploaded also.

SUPERVISOR MEETINGS

Date of Meeting: 28th October

Items discussed:

- Overall project idea
- Technologies involved
- Prototyping
- Upcoming deliverables
- Next steps

Action Items: Get the requirements specification done and ensure that the monthly journals are being completed. Get started on my prototype for the end of semester presentation.

NOVEMBER 2016

Student name: Adam O'Callaghan
Programme: BSc in Computing (Evening) – 4th Year

MY ACHIEVEMENTS

This month I finished and uploaded the completed Requirements Specification for my project. Writing and researching the document has given me a much better overview of my full project functionality and I was also better able to plan out the project architecture and how the elements to the entire system will fit together.

I have also been looking into Natural Language Processing tutorials with Python in order to get to grips with it for the upcoming implementation of the pre-processing backend of my project.

MY REFLECTION

I felt that it helped to properly and deliberately consider the use cases that the software will eventually be used for, as opposed to just winging it and sticking in additional elements and functionality. Previously, I had been considering aspects that I thought would be interesting and clever additions to the system, but on reflection it makes much more sense to purposefully write down and analyse the various functions to decide upon the best ones.

I was hoping to at least get some of the initial coding done, but apart from tutorials this didn't happen. It will be important to start coding up at least the basis of the project throughout December, as a prototype is due for the mid-point presentation and I would also like to have the ball rolling as quickly as possible.

INTENDED CHANGES

Next month I will be focusing on both getting ready for the mid-point presentation and getting my prototype built. For the presentation I will have to ensure that my project concept is delivered in a clear way and that all of the aspects which I feel are interesting and innovative are comprehensively communicated to the examiners.

While a prototype is required for the presentation, I feel that it will also be good to have something implemented and see what direction the project is going, since up until now it has been mainly a paper exercise.