Interim Project Report

Team Correct

Team Members:

Michael Brennan (16200629)

Kunal Chavan (16204154)

Ketaki Abhay Ghatage (16200153)

Xiaosheng Liang (152119133)

Jerom Joy Vattakeril (15200123)

2016-17

Table of content

1 Introduction

After the recent American elections, the issue about fake news (or unreliable, conspiracy, satire, hate, bias, junk, political, clickbait) has surfaced. We have all seen the how spread of fake and conspiratorial news impacted elections in one world’s most powerful countries. There have been many more such instances where people have considered fake or modified news as reality and have responded in a negative way.

The president of the Irish Medical Organisation has said that 'fake news' is playing a role in declining vaccine uptake in Ireland. Dr. Hogan expressed deep concern at the impact of social media campaigns and fake (or fabricated) news about 'non-existent' risks from vaccinations: “Uptake rates for the HPV vaccine amongst young girls are declining to a worrying extent on the back of fake news stories about non-existent risks from vaccinations. "As a result, we are putting the future health of young women at risk of cervical cancer and other ailments” [1].

These news examples are some of the many which have caused impact on politics, health and so on. To prevent people from believing such fake, unreliable, bias, junk, etc. news spread through social media, we decided to build a system which will help reduce the spread and bring people to reality about the news worldwide.

2 User Scenario: The Characters

2.1 Target Users

We can see that impact of fake, unreliable, bias, or satire news is quite high and almost everyone is impacted because of it in some or other way. Hence, our user group consists of:

- *Frequent users*: who check their social media accounts every half hour and share media every day.
- *Average users:* who check their social media accounts every day on an average four times and share media sometimes.
- *Organisations or teams*: who are working on such bigger projects and who can integrate this portion into their system.

At present, there's enormous pressure on firms such as Google, Facebook and Twitter to do more to tackle fake news.

Also, Google is working with organisations like Full Fact, which is developing new technologies that can identify and even correct false claims. Full Fact is creating an automated fact-checker that will monitor claims made on TV, in newspapers, in parliament or on the internet [2].

Technologist Daniel Sieradski has developed a plugin - known as BS Detector - that flags up "questionable" websites on Facebook and Twitter. But they are still looking for more dependable and efficient solutions as it basically uses a list of fake news sources as its reference point. Facebook is looking for more such software model and has already partnered with fact-checking organizations to begin flagging fake news (Figure 1) [3].

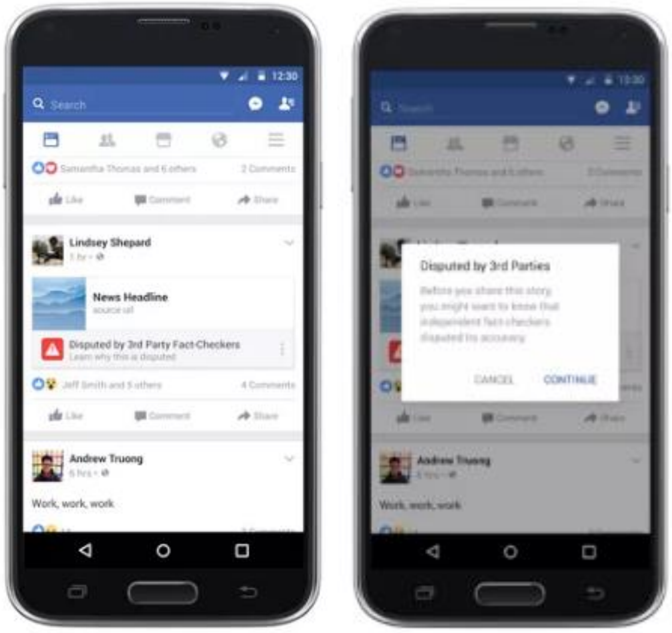


Figure 1 - Facebook flagging

2.3 Why are they important?

There has been a huge impact on people’s opinions due to spread of fake, modified, conspiratorial or biased news. And it is very important that people only see the real (or reliable) news and make decisions accordingly. If we consider the famous controversy over US elections, the results would’ve been very different if people had access to only real news and not fake or modified news. This event speaks for itself about the importance of access to real news.

Today young adults mainly get news from social media, and this trend is on the rise even among other age groups as well. Most users out of these are active throughout the day and share various news articles every hour. Many of those can be fake as well. Research conducted by BBC helps us understand this trend clearly.

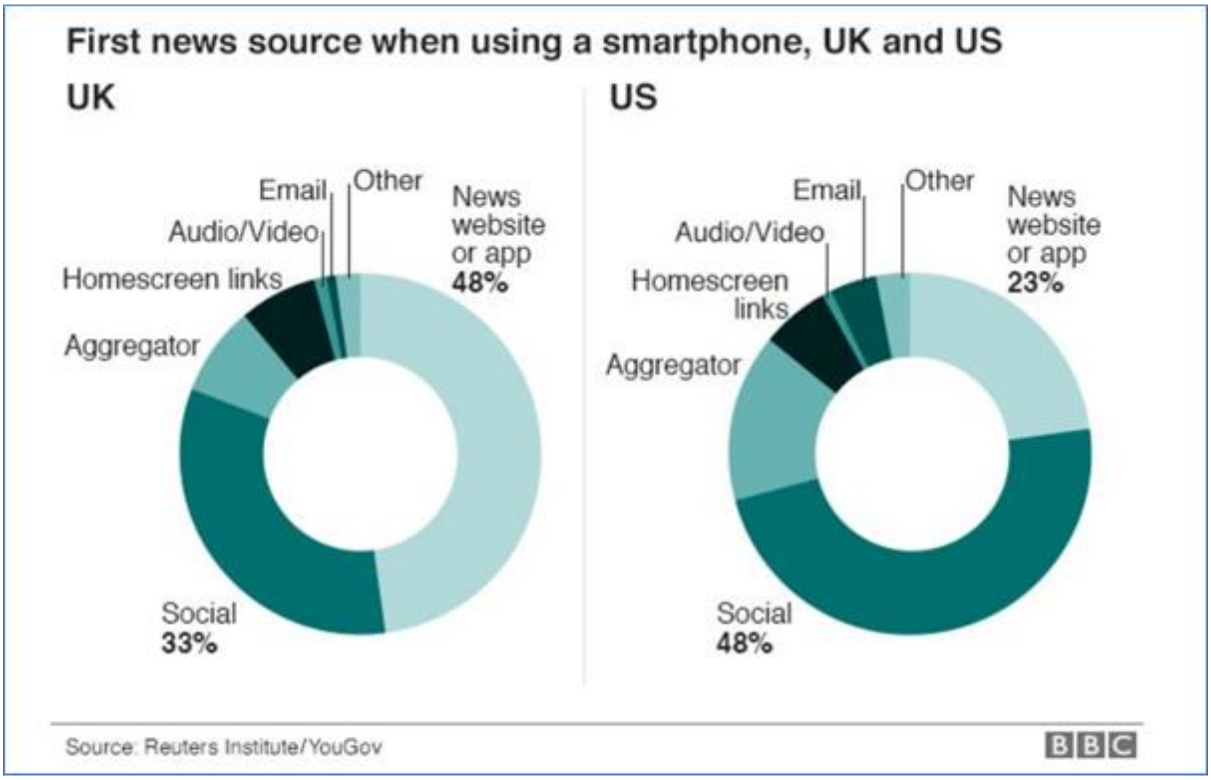


Figure 2 - BBC Social Media Research

Social media has overtaken television as young people's main source of news, according to a report by BBC. Of the 18-to-24-year-olds surveyed, 28% cited social media as their main news source, compared with 24% for TV. The Reuters Institute for the Study of Journalism research also suggests 51% of people with online access use social media as a news source [4] [5].

With firms like Google, Facebook who are looking for such projects to integrate to restore faith among their sharing steam which received a lot of criticism for the spread of fake news.

Such websites have become very important in our lives for mainly socialising and it is important that they maintain their reputation [6].

2.4 What problem are you solving for them?

The news is an integral part of every person's life. It is just not information to be remembered for a couple of hours, instead, it is the information which affects a person's life with respect to how they live, perform our tasks. The news is a compilation of information from different fields hence creating a huge impact if the news is not reliable.

To prevent the users/society from believing unreliable news we are building this website to verify the authenticity of the English language news around the world to come to an understanding that if the news is reliable or unreliable/bias/Satire, etc by comparing our website database with other authentic news sites which are believed worldwide.

Most of the people read news on social media due to the busy schedules leading to a lot of unreal news which creates an unpleasant environment. We are building a system which would help the user understand if the article they are reading is real or not by displaying the news type and reasoning, making it easier for the user to conclude on the news type in comparison to other competitor websites.

3 Technical Problem

3.1 Why does your system exist?

The main aim of our website is to empower people with a tool to safeguard themselves from fake, biased, imposter news sources and help them follow real, unbiased, and official news sources.

Today majority of the population with internet access gets their news from social media, where anyone (official news channels, web channels, individuals) can post news [7]. This exposes people to a variety of information from around the world on a single platform. But, this also exposes them to a threat of being manipulated using fake news [8], which can be through individuals, imposter websites, modified versions of original news, etc.

The website will allow them to tackle this threat as it gives people facility to check authenticity of news articles of their interest. This will be particularly useful for active users as they share news articles they read with other people connected and on an average that number is in thousands. This would be a big group of people and hence, it is important such users check authenticity of a specific news items before sharing. Even a few wrong news items can lead into major controversies. For example, current Gulf crisis, where Qatar is being attacked on basis of some comments by their Emir, which they claim were never said [9].

There are such various instances where controversies could have been averted if proper information was shared in first place. Our website is an attempt to provide help to people in understanding the difference between various versions of real story out there.

3.2 What is the core technical problem?

- *Scraping news from different format websites*

The first step for back-end development in this system is scraping news from reliable and fake i.e. unreliable, conspiracy, satire, hate, bias, junk, political and clickbait news websites. Sites like these are structured on the title, article, author, first image and created date which our system extracts and stored in our database. The data is scraped from several websites and with each one having its own format, as we have experienced this particularly with fake news websites, scraping these collections involves a great challenge in our system [15] [16].

- *Asynchronous Tasks with Celery*

The news is generated every day, hour and minute, our system should be updated periodically to notices these changes to allow our system to deal with the current content. As a large amount of data is stored in our database which needs constantly be updated to provide a real scale rating to the user in an up to date format.

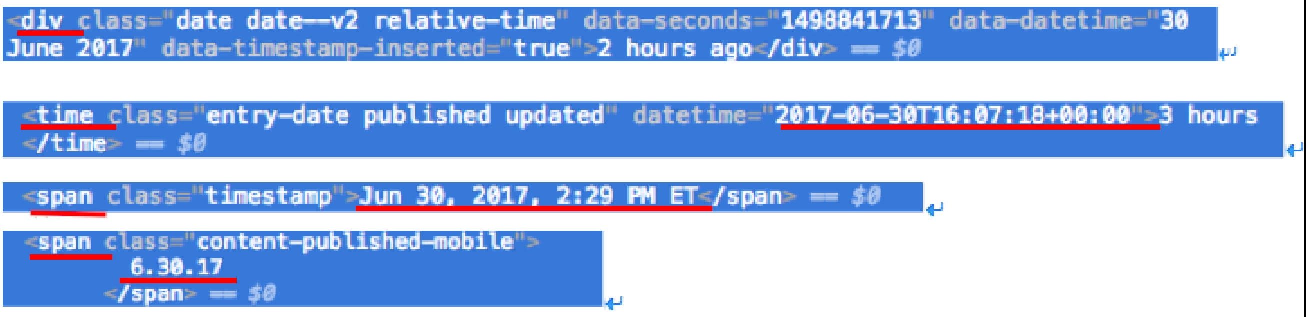
- *News Scoring*

If the user inputs a news' URL in the search bar, the URL will be checked in open source fake news websites list. It would be scored an initial score based on the type of website in the list (figure 4). Then check news' word count in each paragraph, grammar, author, created date and advertisement and get the final score. Each property of news and each type of news should be assigned a specific score based on its portion's weight.

3.3 Can you describe the problem graphically?

- *Web scraping and data format*

Data is extracted from several sources and each source has its own format. Take created as an example, if you inspect the source code of different news web pages, some with created time of news are tagged in "time", some are tagged in "spam", some are tagged in another format. Even all created date is collected, which can be seen like "30 June 2017", some others are "2017-06-30T16:07:18+00:00", "6.30.2017". So, converting to a formal format which can be stored and used in the database is another challenge (Figure 3).



```
<div class="date date--v2 relative-time" data-seconds="1498841713" data-datetime="30
June 2017" data-timestamp-inserted="true">2 hours ago</div> == $0
```

```
<time class="entry-date published updated" datetime="2017-06-30T16:07:18+00:00">3 hours
</time> == $0
```

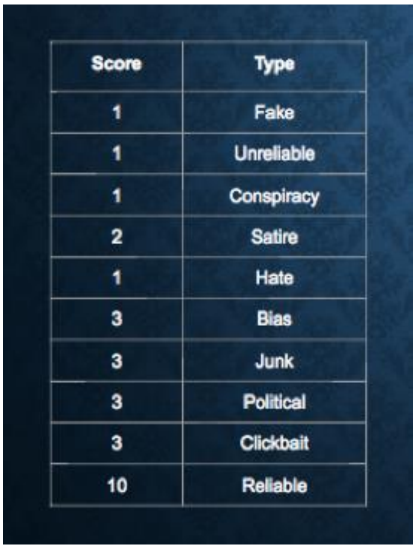
```
<span class="timestamp">Jun 30, 2017, 2:29 PM ET</span> == $0
```

```
<span class="content-published-mobile">
    6.30.17
  </span> == $0
```

Figure 3 - Web Scraping Date Format problems

- *News Scoring*

All news would first be assigned a definite score value base on their domain, e.g. BBC would be one of these domain stores in our database score value of 10 and their domain type base on beginning reliable. Take in consideration of unreliable content with every domain to back up the fact of BBC begin reliable further analysis is done on this example with the article going through a format of identifying the number of words in each paragraph, grammar mistakes, created date, author and advertisements by scraping the article is a real challenge to our system to provide calculate scoring on positive and negative finds. After this process will give the user clear rating on the article and reliable sense of how reliable BBC really is (Figure 4) [11].



Score	Type
1	Fake
1	Unreliable
1	Conspiracy
2	Satire
1	Hate
3	Bias
3	Junk
3	Political
3	Clickbait
10	Reliable

Figure 4 - First Initial Score

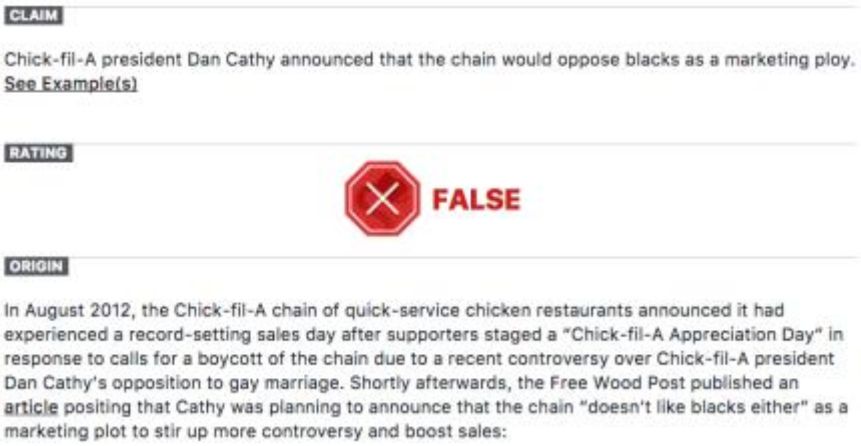
- *Run a Celery worker in background*

Since the programme should be run periodically, Celery should be running periodically. In the development process, when Celery starts, the processing works. Once it stops, the system would terminate. How to run a Celery workers in the background even if the condition of Celery stops is another challenge in back-end development.

3.4 Review other existing systems that address this problem

- *News scoring*

Our system measures news articles with providing a rating score on that article. All news would be scored based on their properties the higher the score the more reliable the news is. But in most of the fact-checking websites like Snopes and FactCheck.org the news is just identified as false or true (figure 5). Also, those websites would allow all user to make claim and provide evidence for a claim at the same time they use the most official, authoritative sources as evidence.



CLAIM

Chick-fil-A president Dan Cathy announced that the chain would oppose blacks as a marketing ploy.
See Example(s)

RATING

FALSE

ORIGIN

In August 2012, the Chick-fil-A chain of quick-service chicken restaurants announced it had experienced a record-setting sales day after supporters staged a "Chick-fil-A Appreciation Day" in response to calls for a boycott of the chain due to a recent controversy over Chick-fil-A president Dan Cathy's opposition to gay marriage. Shortly afterwards, the Free Wood Post published an article positing that Cathy was planning to announce that the chain "doesn't like blacks either" as a marketing plot to stir up more controversy and boost sales:

Figure 5 - Snopes Rating

- *News scraping*

Most news checking websites, like Politifact and Snopes, generate their own fake news. Therefore, all news is displayed in the same format and it's easy to scrape and parse the web page, it's also easy to store all data which are in a standard format [27].

- *Work with task queues*

Task queues manage background work that must be executed outside the usual HTTP request-response cycle. Celery is the most commonly used Python library for handling asynchronous tasks and scheduling [12]. Since Celery is overly complicated for simple use cases, some simply distributed queues are used in different systems. For example, Redis Queue[13] is a simple Python library for queueing jobs and processing them in the background with workers.

4 Technical Solution

4.1 What does your system do?

The system is developed to help users understand the different types of news around the world and to reduce the spread of such news. The system is usable through the front end or the UI (web page). This web page provides a search bar where the user can enter the URL they want to check. This URL is read and checked in the various tables of the Postgres database. The database consists of 7 tables which are used for storing the data. NewsfetchedAPI, Score, Images, News, Sub URLs, DomainFilterList, DomainTypeScore.

These contain the data from NewsAPI as well as the newly added URL information. As news is updated every minute across the world, the system must be updated as frequently as possible and hence update automatically from backend we use Celery. Celery is a worker which helps in running the code automatically. This is done at regular intervals as specified in the code. Word count in the paragraph, grammar check, sub-URLs, Author, Date and advertisements are scored base on the data content which is later totalled to form a final score and help to obtain a news type for the article.

Django is used to run the SQL queries to extract data from the News table and sent to the DomainTypeScore and Score tables to process the final score of the article. The Score and domain type is then stored for the article in the database and sent to the UI to be displayed to the user. The final display of the system to the user would be the Article URL with the score and the type and reason to fall under that domain. If the URL does not exist in the system, we would obtain the URL from the search bar and web scrapes the article and compare the article with other authentic websites and check for its credibility and calculate the score and domain type for the same and store in the database as well as display to the user.

4.2 How does it work?

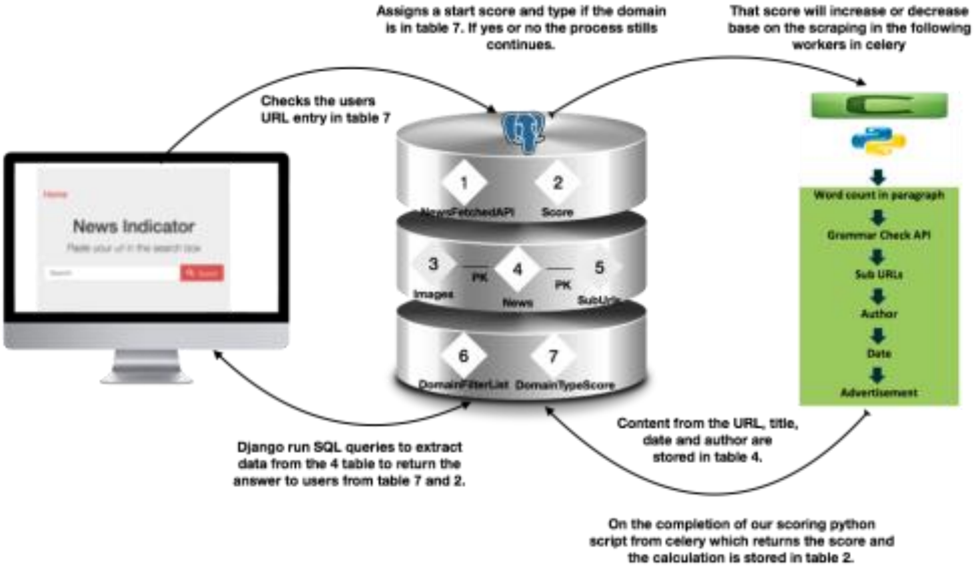

Assigns a start score and type if the domain is in table 7. If yes or no the process stills continues.
That score will increase or decrease base on the scraping in the following workers in celery
Checks the users URL entry in table 7
Home
News Indicator
Paste your url in the search box
Search
1
2
NewsFetchedAPI
Score
3
PK
4
PK
5
Images
News
SubUrls
6
7
DomainFilterList
DomainTypeScore
Word count in paragraph
Grammar Check API
Sub URLs
Author
Date
Advertisement
Django run SQL queries to extract data from the 4 table to return the answer to users from table 7 and 2.
Content from the URL, title, date and author are stored in table 4.
On the completion of our scoring python script from celery which returns the score and the calculation is stored in table 2.

Figure 7 - System Overview

When a user opens the website, on the homepage they can see a search box. There they can enter any news article URL. Once URL is entered in the search box, it is sent to the backend to find any available matches. If a match is found, database (using PostgreSQL) returns news article information (title, content, author, time, and article body) and score of authenticity [17] [18].

In case we do not find a match in our database, using the entered URL, the news article will be scraped (to get title, content, author, time, and article body mainly) and the same will be uploaded in our database for future use. Further, depending on the evaluation of scraped content, a score is allotted to each piece of information. This score is stored in the database as well for future use. Once this is done, the news article information and related score, which is now saved in the database, is displayed on the webpage.

4.3 Front-end

- *Web Framework*

We were decisive on this section with Django or Flask. On much research on these two systems which were new to us, we decided to choose Django, having additional third party packs and larger developer community [19].

- *Programming language*

Python is the best match for Django and two group members having studied the language, hence we feel this would provide a positive start. Psycho2 was selected on it for similarity to SQL studied by four group members.

- *User Interface*

With a big project on a short time frame, we don’t want to be wasting time mocking up CSS tags and screen revolution while bootstrap provided all these elements for us with our own HTML5 coding format.

4.4 How does the front-end works

The front-end works in four parts

1. Django web framework that’s sits on the MVC model.
2. Postgres model which describes the structure of the database table values and types.
3. Psycho2 and Python are positioned in the controller that extract data from the DB by talking to the model by import the model name and query statement or provided to abstract the data in an ordered form.
4. HTML templates are stored in the view to render the UI to the system using CSS and return types from the controller.

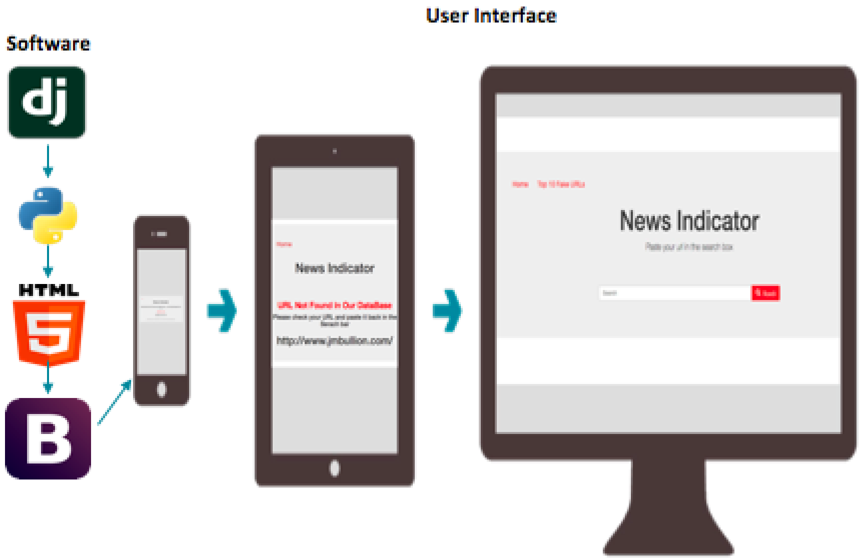

User Interface
Software
HTML
Home
News Indicator
URL Not Found In Our DataBase
Please check your URL and paste it back in the Search bar
http://www.jmbullion.com/
Home Top 10 Fake URLs
News Indicator
Paste your url in the search box
Search

Figure 8 - Front-End System

4.5 Back-end

The website's backend has been mainly developed in Python 3.5. The reason being Python is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python’s elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms. The Python interpreter and the extensive standard library are freely available in source or binary form for all major platforms, for example, requests, beatifulsoup and URL parse.

The above web scraping operations need to be done frequently and on various news sources. Celery implements this in asynchronous tasks and dividing each task as worker gives more control to the processes. Periodic tasks in celery beat make sure that tasks are done in timely fashion and the database is replenished with fresh contents every hour. Even though terminal is good at determining status of the above worker tasks we use Flower to evaluate these tasks and monitor celery with each task progress and history.

A Database with reputation for reliability, data integrity and correctness is necessary for a project and PostgreSQL provides all those features. The high-volume environment support make sure that our ever-growing data will be managed well. As PostgreSQL is primarily made for Unix platform it's well suited for our Linux server and administration and development platform pgadmin4 can be used to visualize data and administration so that data can be evaluated as they are generated from client machines.

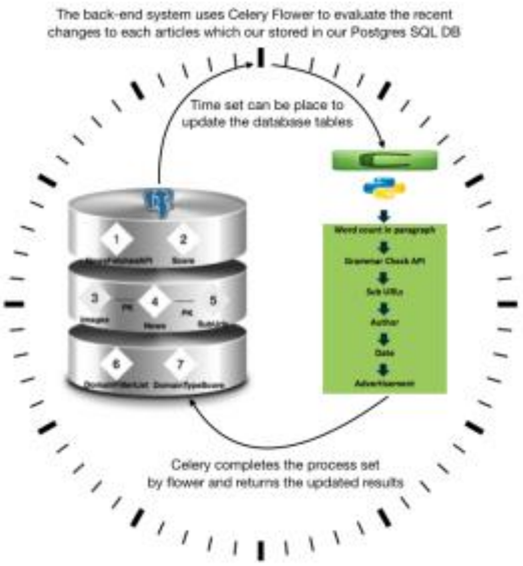
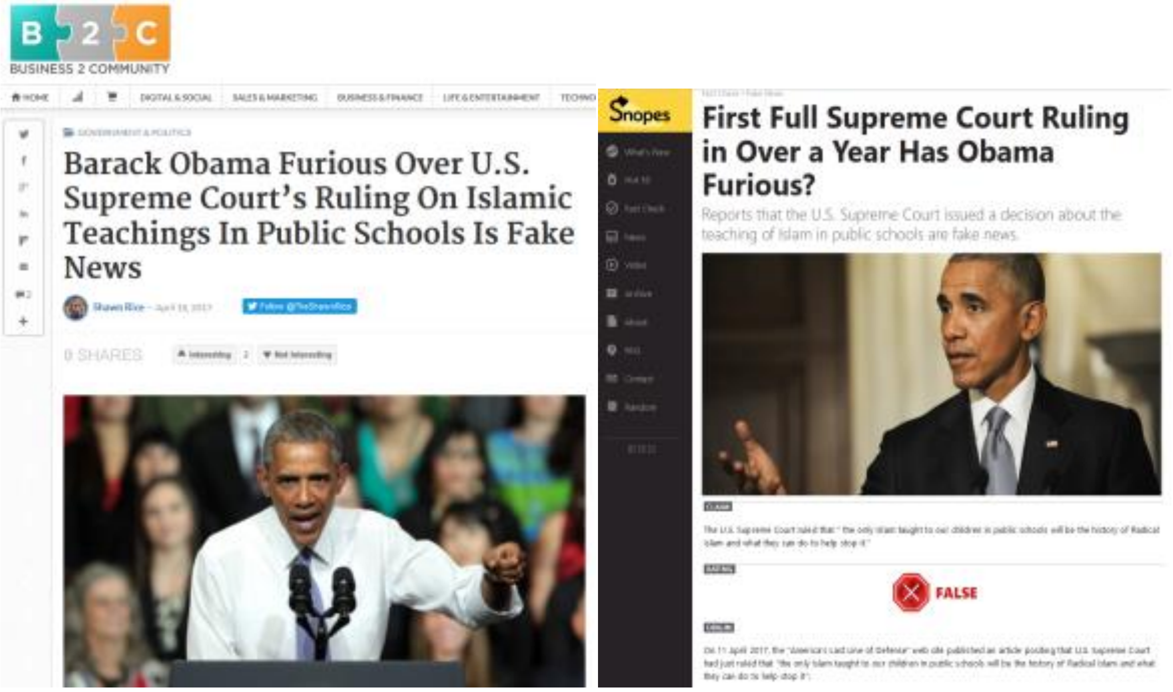

The back-end system uses Celery Flower to evaluate the recent changes to each articles which our stored in our Postgres SQL DB
Time set can be place to update the database tables
1
2
NewsFetchedAPI
Score
3
4
5
PK
PK
imagehl
News
SubUrls
6
7
DomainWhiteList
DomainTypeScore
Word count in paragraph
Grammar Check API
Sub URLs
Author
Date
Advertisement
Celery completes the process set by flower and returns the updated results

Figure 9 - Back-End System

4.6 Data Resources

Our key data resources include:

- *News API for credible news* - We have used an open source API which covers all the credible news sources available. One important feature about this API is that we get the domain for each news item and in some cases, we get secondary domains as well which helps us understand the news items credibility better.
-



Barack Obama Furious Over U.S. Supreme Court's Ruling On Islamic Teachings In Public Schools Is Fake News

First Full Supreme Court Ruling in Over a Year Has Obama Furious?

Reports that the U.S. Supreme Court issued a decision about the teaching of Islam in public schools are fake news.

CLAIM

The U.S. Supreme Court ruled that "the only Islam taught to our children in public schools will be the history of Radical Islam and what they can do to help stop it."

RATING

FALSE

ORIGIN

On 11 April 2017, the "American's Last Line of Defense" web site published an article positing that U.S. Supreme Court had just ruled that "the only Islam taught to our children in public schools will be the history of Radical Islam and what they can do to help stop it".

Figure 10 - An example of fake news detection

In above example, we can clearly see a fake article by business2community website and the same detected by Snopes, which is one of the trusted names when it comes to fake news detection. This is one example of fake news website [26]. There are such hundreds of websites. Our team has made a custom list of such websites which are suggested by reputed websites like Snopes, Fact checker, etc.

- URL Domain filtering using news type - At the initial stage, we collected data (news articles) from various credible sources and kept it at the same level, i.e. no differentiation on basis of fake, reliable, conspiracy etc. This was mainly done to set up a repository which could act as a ground truth for various other articles.

Further we used an API to filter the gathered data on basis of domain and assigned a type to each article. Using Celery, we scraped through them to get required information and stored it in our PostgreSQL database.

Now these domain filtered articles act as ground truth for the various other articles we check and it makes scoring them easier as we are not just relying on scoring of attributes related to only that article (like title, author, date, images, content, etc.)

5 Evaluation

5.1 What does success look like for your system?

As per our study of similar existing systems in the market for fact news websites like FactCheck.org, Snopes, PolitiFact, etc. display the fake news of the USA and no other countries. These systems display a set of news already present in the database and do not provide information as per the user's search.

The features missing in the websites available in the market are taken into consideration and implemented in the software we are developing. Our system is more user-friendly and resolves the issue of knowing about a website, as we provide a search bar to enter the URL of the website and check its content and declaring the results based on the top 4 authentic websites.

We calculate various scores based on the different features like Word count in a paragraph, Grammar Check, Sub URL's, Author, Date, Advertisements of the article. These scores would be available for a website and provide an overall score for the credibility hence, making it easier for users to understand the websites without having to search for a long time. These new features and accuracy will make our system successful than the existing ones in the market.

5.2 How will you evaluate the system that you build?

The core of the system is the news verification. A set of URLs' based on the different categories would be checked by copying the URL in our system as well as other existing systems. If both the systems provide the same result, it indicates our system is on the same line. We would use old news URLs' which is declared true/unreliable/fake type of news and check the URL in 2 different systems and prove our declaration by displaying the score.

The entire system will be evaluated by having categories like Fake, Unreliable, Conspiracy, Satire, Hate, Bias, Junk, Political, Clickbait, Reliable. This is calculated by using different parts of the article like Web scraping, Heading, images, to what extent the news Headings match with other reliable sources, Word Count, Grammar, Date, Author, Sub URLs, adverts. Each of these fields will hold a value as shown below and based on this an average is calculated and displayed using the open source categories available [28] [29].

The categories like Domain, Sub URLs', Advertisements are given 20% priority each. Date and Author fields are provided with 10% each. Grammar is 15 % and word count is 5%, which totals up to 100. The individual score is calculated as follows if the author is mentioned it would be scored 1 if not 0. If the date is recent that is between 0-3 months/3-6/6-9/9-12 months we would decrease the score by 1 starting the score from 5. If multiple grammatical mistakes are found the rating is decreased, the presence of advertisements based on Google is more authentic as per the research made hence we would be scoring it more. Based on our research for advertisement and sub-URLs' that were displayed from websites other than Google was not authentic and were available on the sites which were fake or bias or conspiracy type of news. The values are added and scored and category is declared for it.

6 Testing

The system is evaluated in different phases during the project lifecycle. The main system is divided into three parts which include the Webpage, web scraping and data collection from the URL, database and the connectivity between the 3 parts. Each component is built independently and Unit tested to verify the build works as required.

The web page front end is unit tested by checking the search bar, features like top 10 credible and fake news and the search button click. The web scraping is unit tested by checking the different URL contents scraping, image, advertisement, etc. The database was created and tables were added and the data would be tested. Each component is integrated to the webpage and tested at every stage and a final product testing would be done.

At every phase, a few users would be requested to test the system for its design, quickness, stability and feedback. The feedback would be considered to modify the system and increase efficiency and accuracy at the required zone. If the URL is new to the system and we do not have a pre-existing score then we would add the URL to the system and compare the article with other website and produce the score for the next view (approx. time taken).

7 Conclusion

7.1 Project management strategy

The project management strategy is done through the assistance of a software development methodology. Our workflow is organised into agile method Kanban (Figure 11), which does not include sprites but timely tasks and weekly releases in the following board columns below.

- *Backlog*: Any task which does not get completed within the timeframe set get put into the backlog and would be evaluated if need change would be made by adding or changing member(s) to the assigned task and putting it the highest priority towards the to do. This column we also use for all member to brainstorm idea into a plan of action (to do list).

- *To do*: The task which the project manager will provide, each team member assigned a task compared with their skills sets, this helps to assign the number of members to the tasks also providing realistic due dates also keeping in mind of any issues that may arise.

- *In process*: Start to move into development stage with the monitoring of the project manager and scrum master on a basis to ensure the tasks get to the completed column.

- *Testing*: Series of tests are done with the integration of testing on verification with the assigned team member and our experience team tester member. The following testing is performed, unit, functional, system, performance and acceptance testing before the task gets sent forward for review by our team project owner.

- *Review*: The tasks are compared with the project plan by the project owner validating the assigned task with the scrum master.

- *Completed*: On the completion, the assigned members go back to the start of the process again.

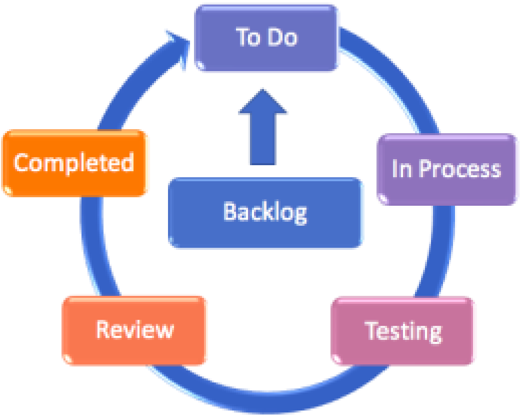


Figure 11 - Agile Kanban

7.2 Project management tools

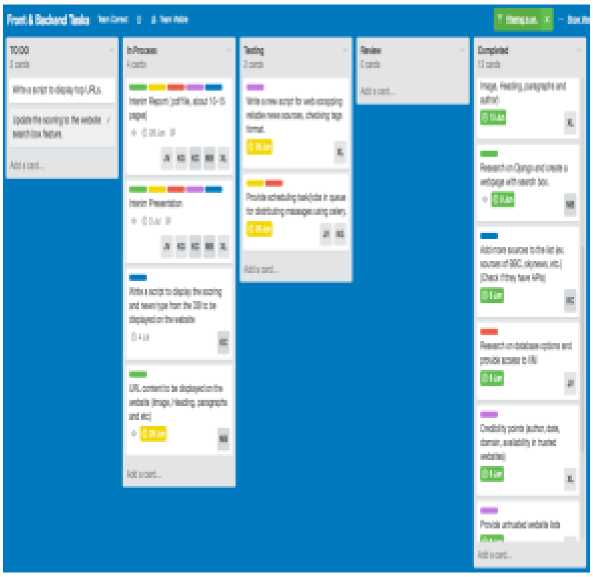
- *GitHub*: The use of this system gives each team member an updated version of the current application also used by all team members to backup code scripts in two folders front-end and back-end.

- *Trello*: Is our whiteboard plans tasks with drag and drop columns of the project workflow with the introduction of the agile approach Kanban (Figure 3). We operate on two boards, board one is all tasks related to the front/backend development of the project, while board two is the documentation deliverables (Figure 1 & 2).

- *Slack*: When the team in different locations the communication is done through slack which gives team real-time notifications on messaging. We use this application to solve troubleshooting, project deliverables and changes being made to tasks.

- *Google Doc*: To keep every team member on the same page when it comes to our documentations and presentations we use this apparatus, with backups of each member's written involvement save on their own UCD email address and on their PC.

- *Google Hangout*: Video conferencing is done mainly at the weekend with group members having part-time jobs at the weekend and family commitments, we find this apparatus great from discussions with the members with different schedules at the weekend to allow brainstorming ideas and stand-ups at an organised time.



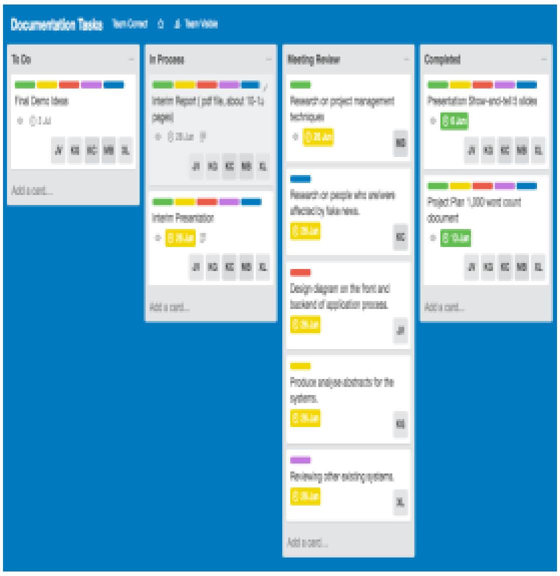
TODO: Write a script to display top URLs. / Update the scoring to the website search box feature.

In Process: Interim Report (pdf file, about 10-15 pages) / Interim Presentation / Write a script to display the scoring and news type from the DB to be displayed on the website / URL content to be displayed on the website (Image, Heading, paragraphs and etc)

Testing: Write a new script for web scrapping reliable news sources, checking tags format. / Provide scheduling task/jobs in queue for distributing messages using celery.

Completed: Image, Heading, paragraphs and author) / Research on Django and create a webpage with search box. / Add more sources to the list (ex. sources of BBC, skynews, etc.) (Check if they have APIs) / Research on database options and provide access to it / Credibility points (author, date, domain, availability in trusted websites) / Provide untrusted website lists

Figure 12- Front and Backend Tasks



To Do: Final Demo Ideas

In Process: Interim Report (pdf file, about 10-1 pages) / Interim Presentation

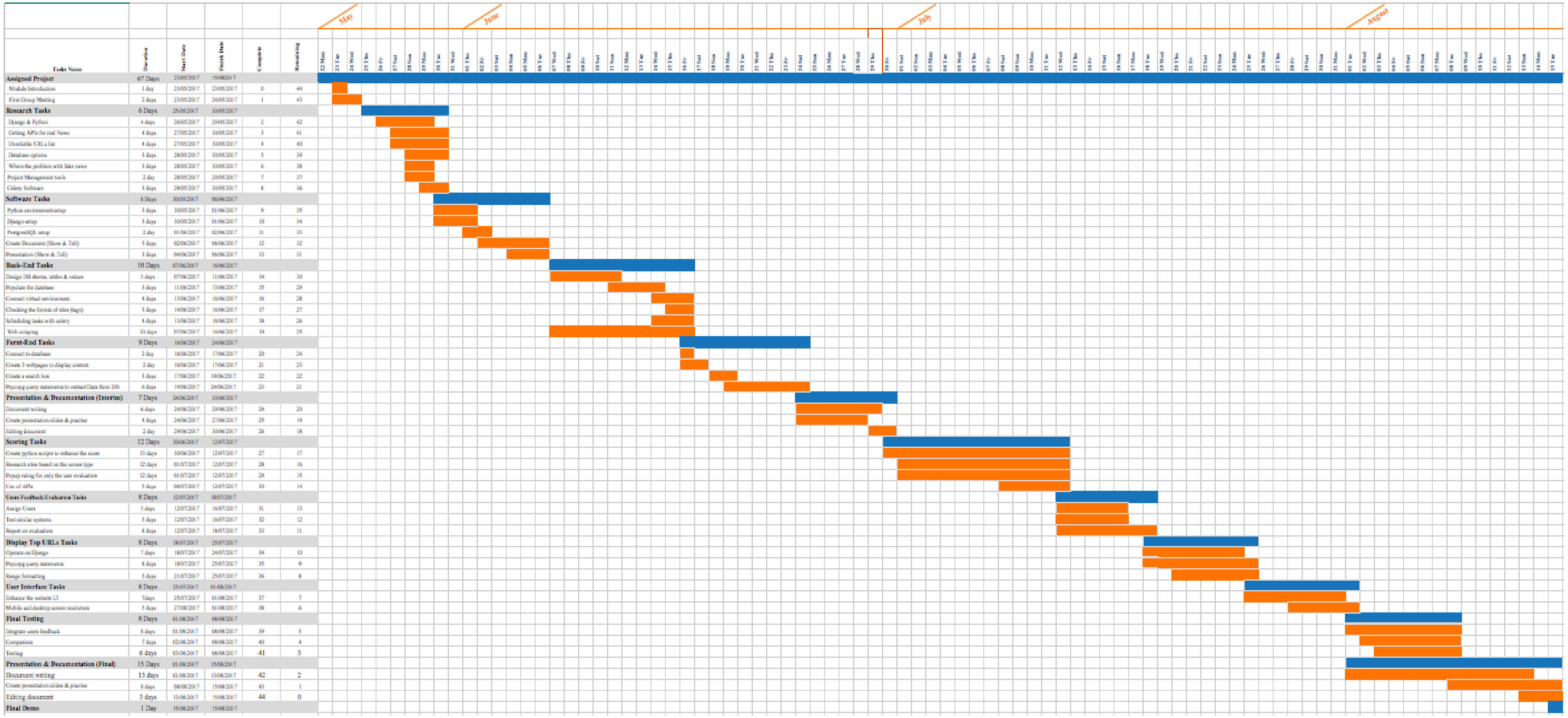
Meeting Review: Research on project management techniques / Research on people who are/were affected by fake news. / Design diagram on the front and backend of application process. / Produce analyse abstracts for the systems. / Reviewing other existing systems.

Completed: Presentation Show-and-tell 3 slides / Project Plan 1,000 word count document

Figure 13 - Documentation Tasks

7.3 Project Timeline

Gantt chart created in Excel (figure 14), gives all members understandable vision of the time remaining.
With the existing 8 weeks, period left our time is framed into several different sections to be completed over a 1 or 2-week stage (highlighted in blue) each section with sub-tasks, (highlighted in orange). No more than 4 sub-tasks to each section is given this frees up remaining members to be formed into the bigger tasks.



Tasks Name	Duration	Start Date	Finish Date	Complete	Remaining
Assigned Project	67 Days	23/05/2017	15/08/2017		
Module Introduction	1 day	23/05/2017	23/05/2017	0	44
First Group Meeting	2 days	23/05/2017	24/05/2017	1	43
Research Tasks	6 Days	25/05/2017	30/05/2017		
Django & Python	4 days	26/05/2017	29/05/2017	2	42
Getting APIs for real News	4 days	27/05/2017	30/05/2017	3	41
Unreliable URLs list	4 days	27/05/2017	30/05/2017	4	40
Database options	3 days	28/05/2017	30/05/2017	5	39
Where the problem with fake news	3 days	28/05/2017	30/05/2017	6	38
Project Management tools	2 day	28/05/2017	29/05/2017	7	37
Celery Software	3 days	28/05/2017	30/05/2017	8	36
Software Tasks	8 Days	30/05/2017	06/06/2017		
Python environment setup	3 days	30/05/2017	01/06/2017	9	35
Django setup	3 days	30/05/2017	01/06/2017	10	34
PostgreSQL setup	2 day	01/06/2017	02/06/2017	11	33
Create Document (Show & Tell)	5 days	02/06/2017	06/06/2017	12	32
Presentation (Show & Tell)	3 days	04/06/2017	06/06/2017	13	31
Back-End Tasks	10 Days	07/06/2017	16/06/2017		
Design DB shema, tables & values	5 days	07/06/2017	11/06/2017	14	30
Populate the database	3 days	11/06/2017	13/06/2017	15	29
Connect virtual environment	4 days	13/06/2017	16/06/2017	16	28
Checking the format of sites (tags)	3 days	14/06/2017	16/06/2017	17	27
Scheduling tasks with celery	4 days	13/06/2017	16/06/2017	18	26
Web scraping	10 days	07/06/2017	16/06/2017	19	25
Front-End Tasks	9 Days	16/06/2017	24/06/2017		
Connect to database	2 day	16/06/2017	17/06/2017	20	24
Create 3 webpages to display content	2 day	16/06/2017	17/06/2017	21	23
Create a search box	3 days	17/06/2017	19/06/2017	22	22
Psycopg query statements to extract Data from DB	6 days	19/06/2017	24/06/2017	23	21
Presentation & Documentation (Interim)	7 Days	24/06/2017	30/06/2017		
Document writing	6 days	24/06/2017	29/06/2017	24	20
Create presentation slides & practice	4 days	24/06/2017	27/06/2017	25	19
Editing document	2 day	29/06/2017	30/06/2017	26	18
Scoring Tasks	12 Days	30/06/2017	12/07/2017		
Create python scripts to enhance the score	13 days	30/06/2017	12/07/2017	27	17
Research sites based on the scores type	12 days	01/07/2017	12/07/2017	28	16
Popup rating for only the user evaluation	12 days	01/07/2017	12/07/2017	29	15
Use of APIs	5 days	08/07/2017	12/07/2017	30	14
Users Feedback/Evaluation Tasks	8 Days	12/07/2017	18/07/2017		
Assign Users	5 days	12/07/2017	16/07/2017	31	13
Test similar systems	5 days	12/07/2017	16/07/2017	32	12
Report on evaluation	8 days	12/07/2017	18/07/2017	33	11
Display Top URLs Tasks	8 Days	18/07/2017	25/07/2017		
Operate on Django	7 days	18/07/2017	24/07/2017	34	10
Psycopg query statements	8 days	18/07/2017	25/07/2017	35	9
Range formatting	5 days	21/07/2017	25/07/2017	36	8
User Interface Tasks	8 Days	25/07/2017	01/08/2017		
Enhance the website UI	5days	25/07/2017	01/08/2017	37	7
Mobile and desktop screen resolution	5 days	27/08/2017	01/08/2017	38	6
Final Testing	8 Days	01/08/2017	08/08/2017		
Integrate users feedback	8 days	01/08/2017	08/08/2017	39	5
Comparison	7 days	02/08/2017	08/08/2017	40	4
Testing	6 days	03/08/2017	08/08/2017	41	3
Presentation & Documentation (Final)	15 Days	01/08/2017	15/08/2017		
Document writing	13 days	01/08/2017	13/08/2017	42	2
Create presentation slides & practice	8 days	08/08/2017	15/08/2017	43	1
Editing document	3 days	13/08/2017	15/08/2017	44	0
Final Demo	1 Day	15/08/2017	15/08/2017		

Figure 14 - Gantt Chart

7.4 Challenges we are currently facing

- *Low engagement*

With members being absent due to home returns, family commitments and having part-time employment covenant with different working hours which cause failure on the discussions around the project and finds themselves to behind on the updated developments discussions.

- *Communication*

Where less than expressed in the week 2 and 3, lack of tools to communication outside the face-to-face meeting to assizes the workflow of group members. Review of this was highlighted in our project plan feedback and research into project management tools took place.

- *Managing expectations*

With a road, mapping uniquely though in the mind of our project manager the vision front end was less bleak affecting the vision of the project.

- *Team skills*

With all members having different language coding skill only two members have experience in python this has heavily but pressure to provide highly challenging Python scripts, if more members had experience with Python this could reduce the challenge.

7.5 How achieve a successful outcome

We have found that collaborating in the same place had helped us achieve our MVP to date, with daily attendance to UCD to work on the project as a team along with stand-ups, and following on with conference video calls on days that are needed.

We expected to face challenges as a team and with taking on individual and team responsibilities throughout the 14-week time frame. Now in week 6 we have learned lots of lessons from these challenges and have overcome them in the milestone of this project, nevertheless to say there still might be forthcoming challenges ahead, if we stay together and communicate at all time and the workflow keeps getting assessed by scrum leader and project manager we will achieve our overall goal.

8 References and Key Resources

8.1 Websites

[1] 'Fake news' behind declining vaccine uptake in Ireland, IMO president warns. (2017, June 30). Retrieved from http://www.independent.ie/irish-news/health/fake-news-behind-declining-vaccine-uptake-in-ireland-imo-president-warns-35645039.html
[2] Facebook Has Introduced a Fact-Checking Alert to Fight 'Disputed Content'. (2017, March 22). Retrieved from http://fortune.com/2017/03/22/facebook-fact-checking-tool/
[3] Facebook is using third-party fact-checkers to fight fake news. (2016, December 15). Retrieved from https://www.engadget.com/2016/12/15/facebook-third-party-fact-checkers-fake-news/
[4] News Use Across Social Media Platforms 2016. (2016, May 26). Retrieved from http://www.journalism.org/2016/05/26/news-use-across-social-media-platforms-2016/
[5] How social media is reshaping news. (2014, September 24). Retrieved from http://www.pewresearch.org/fact-tank/2014/09/24/how-social-media-is-reshaping-news/
[6] Social media 'outstrips TV' as news source for young people. (2016, June 15). Retrieved from http://www.bbc.com/news/uk-36528256
[7] Digital in 2017: Global overview. (2017, January 24). Retrieved from https://wearesocial.com/uk/special-reports/digital-in-2017-global-overview
[8] Bots aren't spreading fake news on Facebook; humans are. (2017, April 28). Retrieved from https://qz.com/971465/facebook-research-paper-bots-aren't-spreading-fake-news-on-facebook-humans-are-fb/
[9] Qatar-Gulf crisis: All the latest updates. (2017, March 22). Retrieved from http://www.aljazeera.com/news/2017/06/qatar-diplomatic-crisis-latest-updates-170605105550769.html
[10] Fake News, Misinformation, and Propaganda (2017, March 28). Retrieved from http://guides.library.harvard.edu/fake
[11] Automated Fact-Checking. (2016, November). Retrieved from https://fullfact.org/automated
[12] How Long Do Users Stay on Web Pages?. (2012, September 12). Retrieved from https://www.nngroup.com/articles/how-long-do-users-stay-on-web-pages/
[13] RQ resource. Retrieved from https://www.fullstackpython.com/redis-queue-rq.html

8.2 Books

[14] Leif, Azzopardi. & David, Maxwell. (2017). How to tango with Django. The Lean Publishing.
[15] Kenneth, Reitz. & Tanya, Schlusser. (2016). The Hitchhiker's Guide in Python. O'Reilly Media
[16] Ryan, Mitchell. (2015). Web Scraping with Python: Collecting Data from the Modern Web. O'Reilly Media

[17] Bruce, Momjian. (2002). PostgreSQL: Introduction and Concepts. Pearson Education Corporate Sales Division
[18] Joshua, Drake. & John, Worsley. (2011). Practical PostgreSQL. O'Reilly Media

8.3 Software/Library

[19] Bootstap. (2017, May 19). Retrieved from http://getbootstrap.com/getting-started/
[20] Python3.6.1. (2017, March 21). Retrieved from https://www.python.org/downloads/
[21] PostgreSQL. (2017, May 19). Retrieved from https://www.postgresql.org/
[22] Django 1.11.2. (2016, December 4). Retrieved from https://www.djangoproject.com/
[23] Postgre Admin. (2017, April 4). Retrieved from https://www.pgadmin.org/
[24] Beautiful Soup.4.4.0. (2017, May 19). Retrieved from https://www.crummy.com/software/BeautifulSoup/

8.4 Papers

[25] Niall J. Conroy, Victoria L. Rubin, Yimin Chen. (2015). Automatic deception detection: Methods for finding fake news. Computer Science. 52 (1), p1-4.
[26] Amadeo J., Torney-Purta J., Barber C. H.. (2004). Attention to media and trust in media sources: Analysis of data from the IEA Civic Education Study. College Park, MD: The Center For Information & Research on Civic Learning & Engagement.
[27] Daniel Glez-Peña, Anália Lourenço, Hugo López-Fernández, Miguel Reboiro-Jato, Florentino Fdez-Riverola Brief Bioinform. (2014). Web scraping technologies in an API world. The Center For Information & Research on Civic Learning & Engagement. P788-797.
[28] Hunt Allcott, Matthew Gentzkow. (2017). Social Media and Fake News in the 2016 Election. NBER Working Paper No. 23089.
[29] Kumaran, G. Gowtham, G. Sudarshan, S.R, Balasubrama, S. (2017). Fake News Detection in Social Media. UCSD CSE PAPER.

8.5 Tutorials

[30] Writing your first Django app. (2016). Retrieved from https://docs.djangoproject.com/en/1.11/intro/tutorial01/
[31] Learn how to query data. (2017). Retrieved from http://www.postgresqltutorial.com/
[32] Bootstrap tutorial on HTML, CSS, JavaScript. (2015). Retrieved from https://www.w3schools.com/bootstrap/
[33] How to use Trello like a pro. (2016). Retrieved from http://help.trello.com/article/734-how-to-use-trello-like-a-pro
[34] Excel Gantt Chart. (2017). Retrieved from http://www.excel-easy.com/examples/gantt-chart.html
[35] Other Fake news detector tools. (2016).Retrieved from http://www.factcheck.org/hot-topics/