



**COMM3790**

**SPECIFICATION REPORT**

# Overview

As identified in the Research Report, the objective of the proposed system is to combat the echo chamber and filter bubble that Facebook's Newsfeed places the user in, as well as the behaviour of constant scrolling and passively consuming the similar posts that appear in the Newsfeed.

The objective is to grab the user's attention in such a way that these behaviours are noticed, and that as a result, a more critical approach to social media is taken.

To do this, the system will take the form of a browser extension and manipulate the data Facebook feeds to the browser, adding an additional post from a database of approved, alternative content. This different, contrasting post will then be presented to the user as if Facebook's Newsfeed had picked it.

This document will present the following:

- A technical overview of how the system is intended to work
- Diagrams of how the system operates and how the user will interact with the system
- Examples of the system in action
- Risk assessment
- Predicted budget and timescale

# Technical Overview

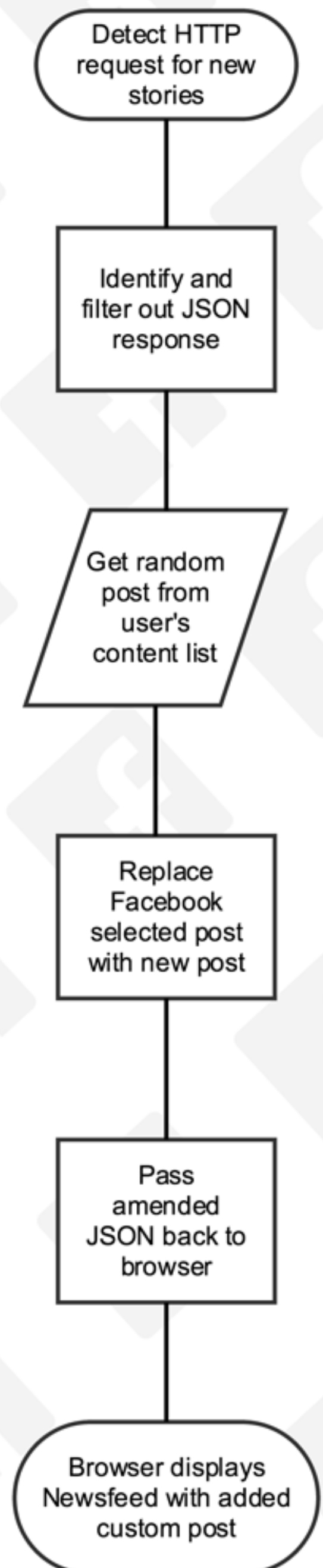
The initial design of the system will take the form of an extension for the Chrome web browser. When downloaded, an initial setup will run (covered later in the report) from which basic settings will be determined for the user. After this, the extension runs in the background with no further input required from the user.

While enabled, the browser waits for a specific HTTP request Facebook makes to gather more stories. This happens when a user is scrolling their Newsfeed and takes the form of:  
<https://www.facebook.com/ajax/pagelet/generic.php/LitestandMoreStoriesPagelet?>  
with a UTF-8 encoded string of data appended.

The response comes in JSON format, the majority of which is irrelevant to this report. Within this is a formatted list of content to display as posts in the Newsfeed. The structure of this content is relevant and examples of data structures and the resulting posts can be found on the next page.

The system would then select a random piece of structured content from a user-selected 'content list' and insert this into the large JSON response.

This response would then be passed onto the browser and displayed in the Newsfeed.



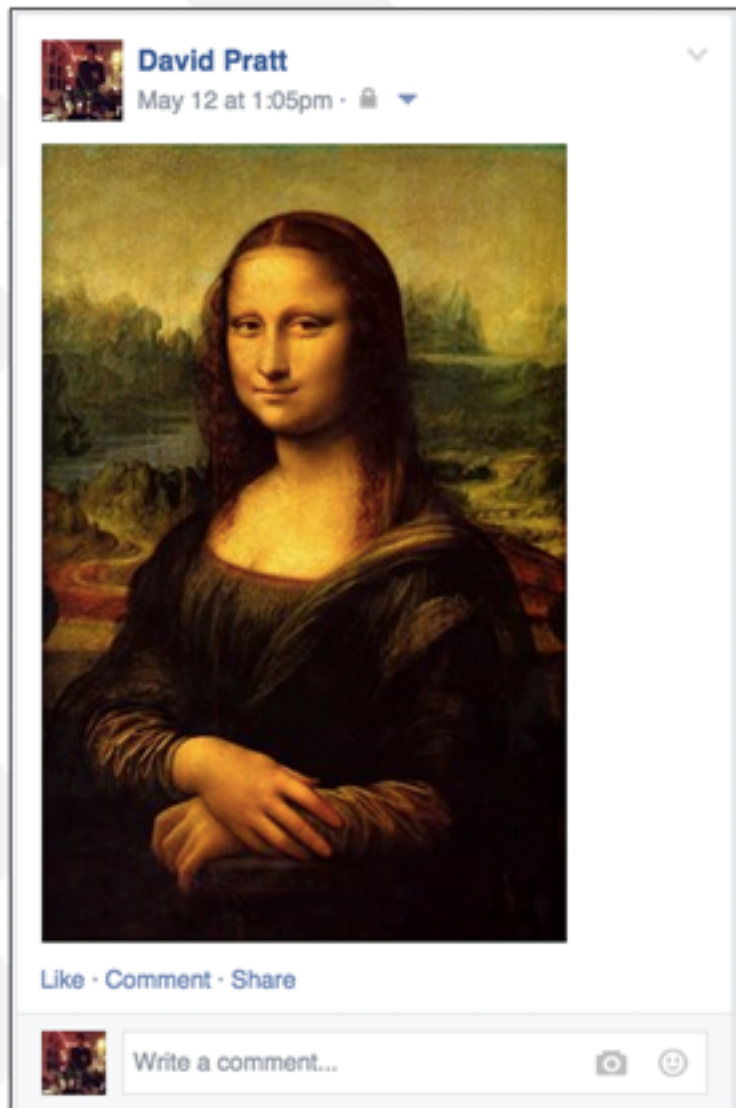
# Technical Overview

## Facebook Status



## Code Structure

```
{
  "id": "1450227118568931_1583718205219821",
  "from": {
    "id": "1450227118568931",
    "name": "David Pratt"
  },
  "message": "This is an example of a text-based status.",
  "actions": [
    {
      "name": "Comment",
      "link": "https://www.facebook.com/1450227118568931/posts/1583718205219821"
    },
    {
      "name": "Like",
      "link": "https://www.facebook.com/1450227118568931/posts/1583718205219821"
    }
  ],
  "privacy": {
    "value": "ALL_FRIENDS",
    "description": "Your friends",
    "friends": "",
    "allow": "",
    "deny": ""
  },
  "created_time": "2015-04-28T23:57:30+0000",
  "updated_time": "2015-04-28T23:57:30+0000",
  "is_hidden": false,
  "subscribed": true
}
```



```
{
  "id": "1450227118568931_2199006046813240",
  "from": {
    "id": "1450227118568931",
    "name": "David Pratt"
  },
  "picture": "https://scontent.xx.fbcdn.net/hphotos-xpa1/v/t1.0-9/s130x130/11263072.....",
  "icon": "https://www.facebook.com/images/icons/photo.gif",
  "actions": [
    {
      "name": "Comment",
      "link": "https://www.facebook.com/199098633470668/posts/2199006046813240"
    },
    {
      "name": "Like",
      "link": "https://www.facebook.com/199098633470668/posts/2199006046813240"
    }
  ],
  "privacy": {
    "value": "ALL_FRIENDS",
    "description": "Your friends",
    "friends": "",
    "allow": "",
    "deny": ""
  },
  "type": "photo",
  "status_type": "added_photos",
  "object_id": "2199006046813240",
  "created_time": "2015-05-14T22:05:41+0000",
  "updated_time": "2015-05-15T00:26:11+0000",
  "shares": {
    "count": 1091
  }
}
```

# User Experience

The system is designed to require as little user input as possible. Considering it runs passively in the background, for a casual user, the only decisions that need to be made is deciding what content list to subscribe to, which will determine what sort of alternative posts show up in the Newsfeed. This simple design philosophy is inspired by the Research Report, which made the point of user comprehension falling when presenting with a mass of information and options.

## Subscribe to a Content List

---

A content list contains all of the posts that can be picked to insert into your Newsfeed.

**Verified content lists**

- Subversive [102 users]
- Shocking [88 users]
- Bubble-bursting [80 users]
- UnfilterMe [70 users]
- Freelist [49 users]

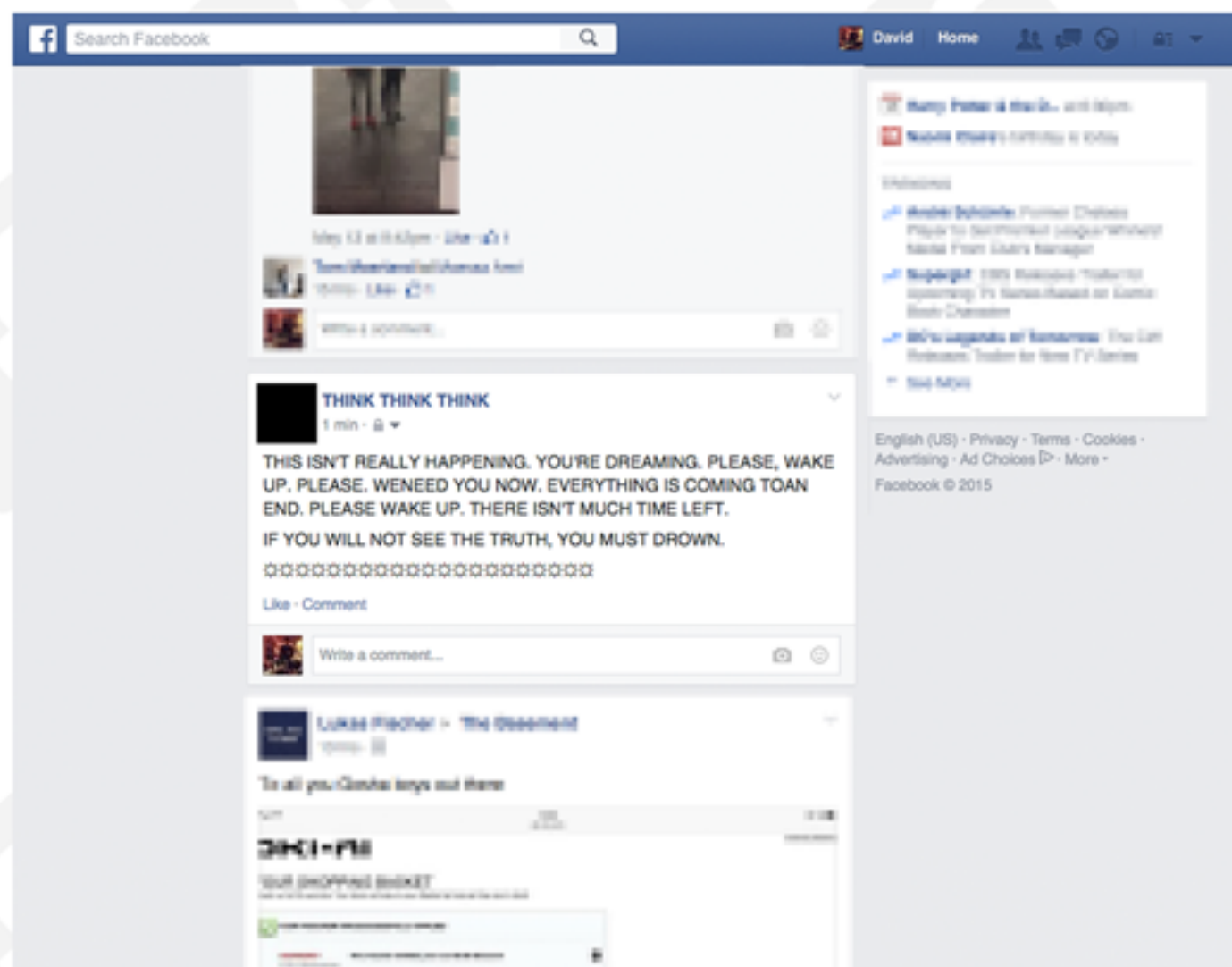
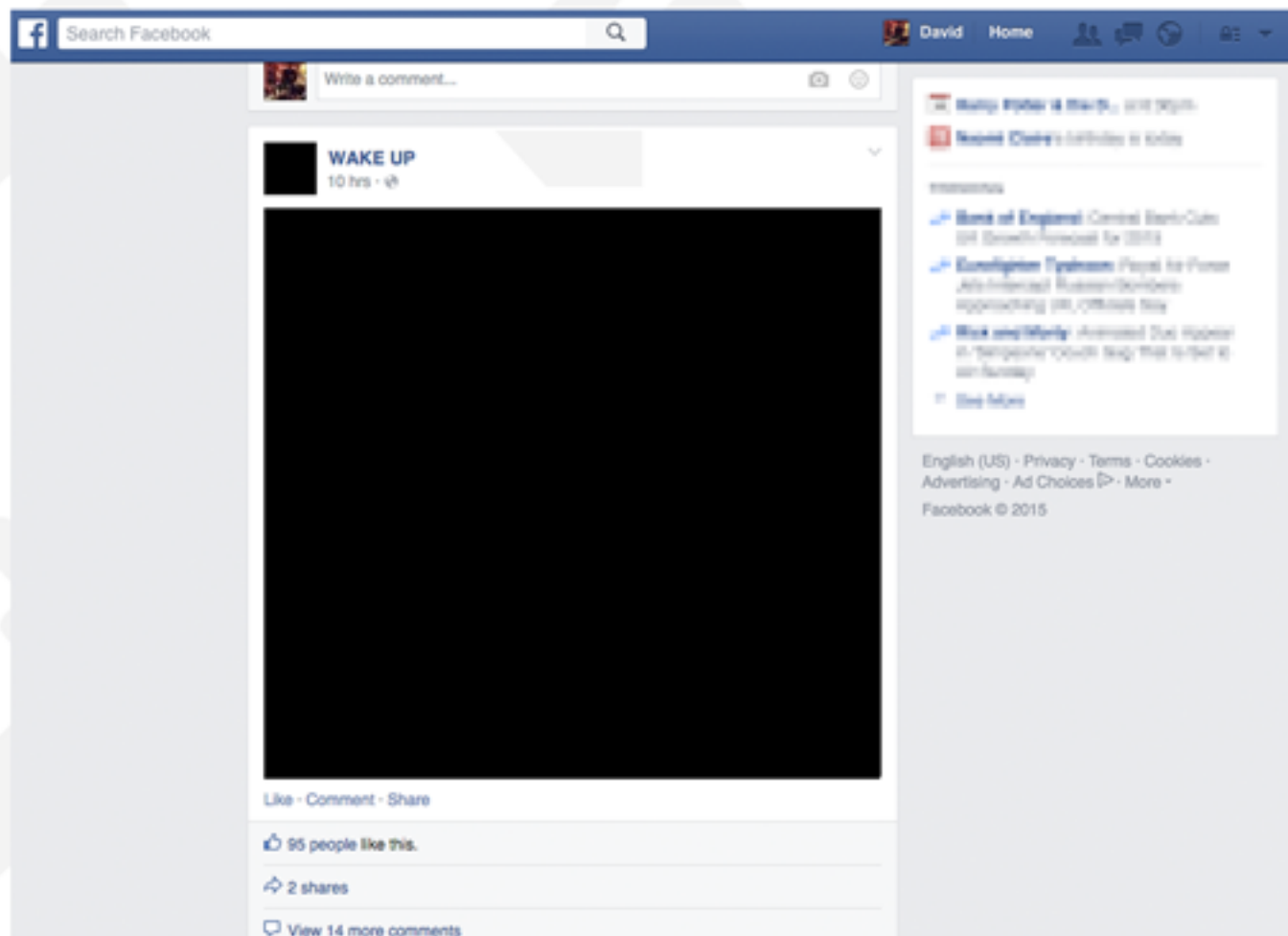
**Custom content list (URL)**

After the user installs the extension from its page on the Chrome web store, they will be presented with a simple form where they can decide which lists to subscribe to. The form will also allow the user to upload their own content list (which will be discussed later). After this has been done, the extension will be running in the background, and the user will begin to experience alternative posts in their Newsfeed.

# Demonstration

For the initial version, types of content allowed on lists will be limited to text and images, although there is technically nothing to stop videos and even remote javascript being used to have an even greater effect on the user.

Below are some screenshots of what a user's Newsfeed could look like when running the extension.



# Content Lists

Content lists are the heart of the extension as they hold the alternative content that gets displayed in the user's Newsfeed. As mentioned in the Research Report, one problem this system seeks to overcome is introducing another 'editorial voice' that selects content for the Newsfeed. As a result, these content lists can be created and used by anyone.

In terms of data, these lists would store the content in the same format that Facebook does. In order to lower the technical barrier of creating a list, a simple web service will be provided that allows the user to provide what content they want on a list in simple, rich format, which can then be parsed and converted into JSON format.

These lists will then be stored on a dedicated hosting server

## Content List Creator

---

The content list creator currently only supports text and image posts.  
For a text-only post, just fill out the textbox. For an image-only post, just upload an image. For both, fill out both.  
Click 'Add' to add the post to your content list. When you're done, press 'Finish'.

Status Text:

Image:  No file chosen

# Risk Assessment

By adopting the strategy of keeping the project as lean as possible, the scope for error decreases. It is also hard to comment on risky externalities before properly understanding how the system will operate when live. That said, there are a few immediate aspects that require attention:

**Hacking/System Exploit** - There is the possibility of malicious code being uploaded through the content list creation system. While the upload form will be as secured as possible with all inputs sanitized, such a black swan can't be ruled out. As a precautionary principle, no sensitive data will be stored on the server, and regular penetration testing can be undertaken.

**Illicit Content Uploaded** - The philosophy behind the project is one of increasing freedom and free speech, but there are boundaries to such principles. Transparent, light-handed moderation will be used to make sure no illegal content is present in the system. This will also prevent spam.

**Lack of Use** - The more people using the system and amending content lists, the better it will work. Spreading the message in appropriate networks will encourage uptake, as will constant maintenance of content lists by those initially involved.

**Browser Incompatibility** - The project will be open sourced, allowing for transparency, code improvement, and also allowing others to port the work to multiple platforms.



# Budget & Timescale

There are two main points of action: coding the system, and designing initial content.

Popular freelance website elance.com points towards a competitive rate of £15/hr for a Chrome extension programmer. It is my belief that they can be given this brief and have a working system implemented and tested - including the content list system - within 20 hours. This puts the price for the initial creation of the extension at around £300.

As mentioned in the Risk Assessment, the quality of content lists is of utmost important, and to this extent I would want to hire a professional designer. elance.com cites a competitive rate of £30/hr. To launch, I would want at least one content list containing 20 high quality posts. Ballparking, I would put this at around 60 man hours, for a total of £1,800.

This puts the cost for the initial creation and launch of the system at £2,100, and a time-to-launch of around a week.

That being said, I anticipate the philosophy of the project to appeal to several internet communities that would be willing to contribute their ideas for free. I also believe that this would be the better option in the long run due to the added bonus of having people of a similar philosophy engaged in the project being able to contribute ideas.

I believe that attracting a small group of hobbyists to contribute to the project would be the best option in the long run, in terms of both budget and timescale. Not only will it cost less, but it is likely to be the best option for the longevity and growth of the project. Having a team that understands the

project's philosophy and are constantly updating it is preferable to a freelance programmer who has to be paid hourly.

In doing this, the full ambitions of the project can be realised. That is to say that encouraging a user to take a critical stance of their social media usage can be done across multiple websites. Not only this, but by allowing multiple voices to be present within the same medium will prevent the creation of echo chambers, which in turn will hopefully allow social media to benefit society in the ways initially discussed in the Research Report.