uc3m | Universidad **Carlos III** de Madrid

Master Degree in Cybersecurity

2020-2021

Master Thesis

"Ad Blocker implementation using HTML code parsing and modification in websites"

Diego García Garín

Ángel Cuevas Rumin

Ad Blocker implementation using HTML code parsing and modification in websites

Diego García Garín

Master Degree in Cybersecurity

2020 - 2021

Universidad Carlos III de Madrid

Abstract—**The main objective of this project is to develop an extension for Google Chrome browser capable of detecting and blocking advertisements on web pages by parsing and modifying the HTML code of those pages.**

Extensions are pieces of software that are installed in the browser and perform similar to the applications of your smart phone, fulfilling specific functions that improve the user experience.

The development of the project has been divided into two main sections: the creation of an extension for the browser of Google Inc. (Google Chrome) using JavaScript programming language and JavaScript Object Notation, an extended text format for data exchange. And a second section where an ad detection algorithm has been developed, tested and improved, performing multiple tests of various websites with a variety of ads.

The end result is an extension capable of detecting the vast majority of ads displayed by the tested websites and minimizing the removal of legitimate code from the application.

Keywords— Extension, Advertisements, Web pages, Google Chrome, AdBlock, JavaScript.

I. INTRODUCTION

For as long as there have been products to market, there has been a need to communicate their existence; the most common form of advertising was oral expression. The earliest example of advertising was found in Babylon, a clay tablet with inscriptions for an ointment merchant, a scribe and a shoemaker dating back to 3000 B.C. Since then, advertising has evolved steadily over the years and across cultures. Advertising as it is known today began to emerge in the United States and Great Britain at the end of the 18th century during the industrial revolution, where specialized advertising agencies began to emerge. In Spain in 1872, advertising pioneer Rafael Roldós founded the country's first agency in Barcelona, which is still active today. [1]

Advertising as it is understood today has two main objectives, firstly and ideally, advertising informs the consumer about the benefits of a particular product or service, highlighting the difference over other brands and secondly, advertising seeks to bias by psychological means the motivational balance of the certain subject towards the advertised product, so as to increase the likelihood that the consumer acquires the object or service advertised.

Today, the main form of advertising is through digital media such as social networks and the Internet in general where it is estimated that about 60% of the world's population has access and where, on average, users remain connected 6 hours 54 minutes a day. [2] [3]

It is for this reason that advertisers spend a lot of money to reach potential customers. In the United States alone, advertising consistently accounts for around 1% of gross domestic product (GDP). This year, global advertising across all media is forecast to reach half a billion dollars, with digital advertising surpassing TV spending for the first time, at 40% versus 36% (and expected to reach 50% by the end of this year). [4]

And while the use of paid advertising to subsidize content is nothing new, the medium is changing. In particular, with the shift to digital advertising, consumers are seeing more and more ads, and in many cases, lower quality ads, and those ads have introduced new problems; more specifically, users on average receive between 3,000 and 5,000 advertising messages per day. [5]

The dramatic increase in advertising on web pages is what has led to the development of ad blockers. Ad blockers have been around since at least 2002, and their use continues to increase as the tension between content, user experience and advertising shows no signs of abating. For many users, ad blockers have made the Web readable again.

However, the advertising industry was not about to sit idly by with the increased development of these ad blockers so advertisers have developed methods to block the operation of the most popular ad blockers on the market.

The aim of this project is precisely to tackle this problem so that users do not have to face a disproportionate amount of advertising every time they want to access a website that has implemented mechanisms for the detection of the most common ad blockers.

In this article it will be discussed the state of the art, in short, the alternatives that exist today in the market to avoid the appearance of advertising, and how this project can fit in this environment.

In addition, it will be explained the challenge of implementing a tool of these characteristics, and what has been the solution that has been reached to solve these challenges.

It will also be explained how the solution has been implemented and finally the results obtained in the tests carried out on different websites will be presented.

II. STATE OF THE ART

This chapter aims to give an overview of the concepts that will be discussed in this project, starting by explaining why Google Chrome browser has been chosen instead of some of the alternatives, then it will explain what is a web extension and finally the different alternatives that exist in the market today for ad blocking.

A. Web browser

In order to decide which browser to use, a study of which are the most used browsers has been carried out with the objective that most of the users can make use of the developed tool.

A web browser is a software application for viewing websites, documents and data. The most popular current browsers are Microsoft Internet Explorer, Google Chrome, Mozilla Firefox and Apple's Safari. Historically, one of the big players in the segment, Internet Explorer, seems to have lost its importance on the web browser market.

According to data provided by Statcounter, Google Chrome has been the most popular browser in the United States since December 2013. In other countries, Google Chrome has also been the dominant browser. In the European browser market, Chrome and Firefox have established solid market positions with about 60.7% and 12%, respectively. The Canadian browser market, however, shows a slight variation, with Safari being the second force behind Chrome as of Q2 2021. Globally, Chrome provided a share of more than 65% of the global web browser market as of June 2021, making it the most used browser in the market by a significant margin as shown in the chart below: [6]
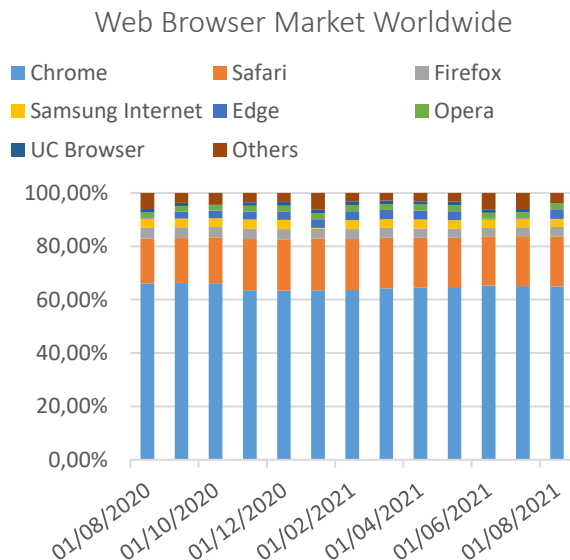

Figure 1. Web Browser Market Worldwide

Another reason for choosing Google Chrome browser is because of its ease of creating and incorporating extensions. Google Chrome has an extension marketplace that offers a huge amount of free extensions created by the community, in fact any developer with enough knowledge is able to create and share an extension in this marketplace.

B. Extension

Extensions are similar to applications that are installed on the browser and that, like applications on a mobile device, perform specific functions that, among other things, improve the user experience.

When Google Chrome is installed on a computer, certain extensions that a common user use on a daily basis such as mail, Google Search, Google Docs, Google Sheets or Google Presentations are automatically installed. However, there is a wide variety of free extensions that can help users to increase their productivity, improve their online security, block advertising, among many others.

Technically, extensions are based on web technologies such as HTML, JavaScript and CSS. They run in a separate and isolated execution environment and interact with the Chrome browser to extend its capabilities through the use of APIs (Application Programming Interface) to modify the browser's behaviour and access web content.

For been able to use extensions in a web browser there are two main elements to take into consideration: a user interface and an API for developers:

The extensions **user interface** provides a consistent way for users to manage their extensions, being able to add, modify or remove them from their browser.

The extensions **APIs** allow the extension code to access features of the browser itself: enable tabs, modify network requests, and so on.

As mentioned earlier, in addition to the extensions that can be found on the market, it is also possible to create extensions from scratch and for this, it is necessary to have a number of resources to do so, which are: a manifest, JavaScript and HTML files, images, etc. More details on this issue will be provided later in this report.

C. Adblocking

Today there are several alternatives to block advertising and most likely, many of the readers of this article are already using some of them.

In fact:

- **42.7%** of internet users worldwide between 16- and 64-years old use ad blocking tools at least once a month.

- "AdBlock", a popular blocking extension, is reported to have over **65 million** users.

- Blocking ads is most common among internet users **between 16 and 24 years old**, **46.2%** of this demographic group worldwide use ad blockers.

Worldwide, the most frequent reasons for using ad blockers are the **excessive number of ads** (22.3%), the **irrelevance** of advertising messages (22.3%) and the **intrusiveness factor** (19.9%), as shown in the table below: [7]

Reasons for usign ad blockers

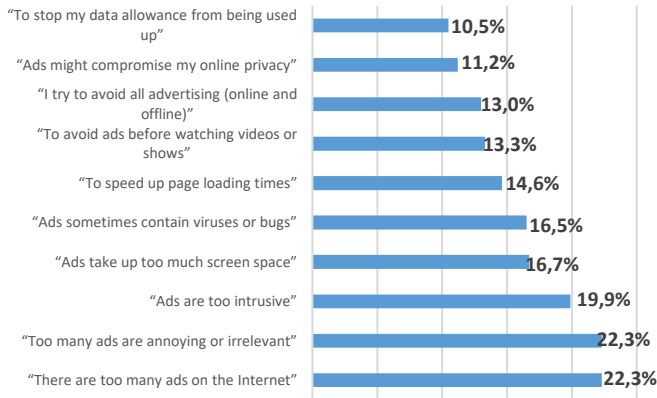| Reason	Percentage
"To stop my data allowance from being used up"	10,5%
"Ads might compromise my online privacy"	11,2%
"I try to avoid all advertising (online and offline)"	13,0%
"To avoid ads before watching videos or shows"	13,3%
"To speed up page loading times"	14,6%
"Ads sometimes contain viruses or bugs"	16,5%
"Ads take up too much screen space"	16,7%
"Ads are too intrusive"	19,9%
"Too many ads are annoying or irrelevant"	22,3%
"There are too many ads on the Internet"	22,3%

Figure 2. Reasons for using ad blockers

As said before, the most popular of them all is: "AdBlock", but it is not the only one, there is also AdBlock Plus, uBlockOrigin, among others. And all of them are based on the same principle, **block advertising before it reaches the user's browser**.

What does this technique consist of?

Strange as it may seem, conventional ad blockers do not block ads, but rather block requests to download ads. Here's how it works: when someone visit a website, for example, the home page of youtube.com, the browser downloads that web page to your computer in parts, as it requests the individual pieces of text, images and underlying code that make up the page, i.e., HTML file, CSS, JavaScript, PNG, JPG, etc. On many sites, some of that content is advertising from third-party ad servers. As the browser builds the page, the ad blocker checks each request for calls to those third-party ad servers. When it finds a match, it blocks the request from loading in the browser. [8]
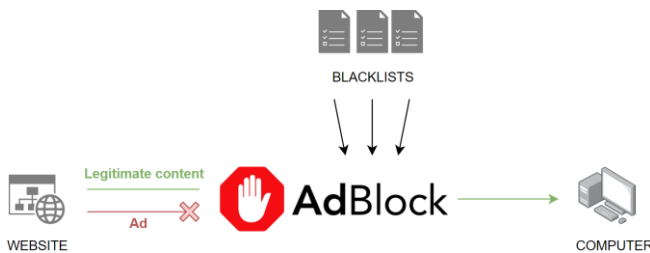

Figure 3. AdBlock flow

This checking of each request is done using so-called filter lists or blacklists. These lists are not part of the extensions but are independent of the extension and generated by the Internet community. These lists contain a set of expressions, rules and filters that correspond to advertising content. Ad blockers compare each request with the entire list and if they find a match they block the request, thus preventing the browser from loading the advertisement. [9]

However, this method of ad blocking can be detected by websites and force the user to disable the ad blocker.

One of the techniques used to detect ad blockers is by exploiting ad blockers using block lists that include a reference to "wp-banners.js", which is a common name for JavaScript files that are associated with ad serving. Knowing this, a JavaScript code is generated that creates a hidden <div> in a file called "wp-banners.js" and placed in the root directory of the website.

Subsequently, after the browser has loaded the web page, it checks whether the hidden <div> created with the "wp-banners.js" file exists (ad blocker off) or not (ad blocker on). [10]

III. PROBLEM DEFINITION

As explained above, existing ad blockers use blacklists to block requests containing ads from loading in the browser, however this technique can be detected and therefore the content of a web page can be restricted to users using ad blockers.

The goal of this project is to create an ad blocker that is able to circumvent existing ad blocker detection techniques and at the same time block as many ads as possible.

This problem has been approached by using ad detection, modification and parsing techniques in the HTML code of web pages. In this way, ads are actually loaded by the browser, but are hidden or removed before being displayed to the user.

Figure 4. Developed extension flow

The main challenge of implementing this technique arises when it comes to detecting ads in the HTML code of web pages. Each web page has been developed with its own particularities, characteristics and unique nomenclature, so the task of detecting which part of the HTML code corresponds to an advertisement and which is not.

Before discussing how this problem has been addressed and what has been the solution that has been reached, it is important to know in depth how web pages work.

Most web pages are composed of two elements: the frontend and the backend. The frontend is the part of the web page that a user can access directly through his computer. It is the set of web design and development technologies that run in the browser and are responsible for interactivity with users.

Among the technologies that make up the frontend, the following can find: HTML and CSS, the layout languages that allow developers to define the structure and styles of a web page, and also JavaScript, a programming language to define the logic of the web page and that receives user requests and send them to the backend to be processed.

The backend is the data access layer of a software or any device and contains all the logic necessary to use this data and is not directly accessible by users. The backend also has access to the web page server, which is nothing more than an

application or piece of software that is responsible for processing the requests made by the browser.

Browser extensions run on the client side, which means that they use frontend technologies to run and it is precisely this set of technologies that will be used to block advertising:

- **HTML:** (HyperText Markup Language), is the language where the information or content of the document is defined.

- **CSS:** (Cascading Style Sheets), is the language where the design of the document is specified, it handles everything related to the visual part

- **JavaScript:** the programming language that allows instructions to be given to the browser to execute certain tasks that give web pages the ability to be interactive.

In order to detect advertisements, it is important to know the structure of HTML files:

HTML uses "markup" to tag text, images and other content for display in a web browser. HTML markup includes special "elements" such as <head>, <title>, <body>, <header>, <footer>, <article>, <section>, <p>, <div>, <span>, <img>, <aside>, <audio>, <canvas>, <datalist>, <details>, <embed>, <nav>, <output>, <progress>, <video>, <ul>, <ol>, <li> and many others, where each one specifies what content is to be shown to the user or how that content is grouped.

Every HTML document consists of the following 3 elements:

- **The <html> tag:** is the root element that defines the entire HTML document.

- **The <head> tag:** contains meta information such as the title of the page, the character encoding used on the page, etc.

- **The <body> tag:** encloses everything that will be displayed to the user.

On top of that, there are three main parts of an element, that are as follows:

- **Opening tag**. This is used to indicate where an element begins to take effect. This is wrapped with opening and closing brackets, for example, <body>.

- **Content**. This is the content to be displayed to the user.

- **Closing tag**. This is represented the same as the opening tag, but with a slash before the element name, for example, </body>.

These elements can have a number of attributes that are defined in the opening tag and consist of two sections: a name and a value. There are many attributes for each element but this document will not go into detail about each of them, however, it is key to explain two of the most important attributes that can be found in any element of the document, which are the id and the class:

- **Id:** is the attribute that defines a unique identifier (ID) for a certain element and that must not be repeated in any other part of the document. Its purpose is to identify the element when it is to be

used in scripts (with JavaScript) or style sheets (with CSS).

- **Class:** defines a class for a given element, and there may be more than one element with the same class. Often classes are used to define the same style for a set of elements, instead of defining the same style for each element separately. [11] [12]

The following images show an example of HTML code where you can see the three main elements mentioned above, as well as an example of the use of the class attribute. Followed by an image of the resulting web page, that is, what is shown to the user:

```html
<!DOCTYPE html>
<html>
<head>
<style>
.note {
  font-size: 120%;
  color: red;
}
</style>
</head>
<body>

<div class="note" style="border: 1px solid black">
Hello World</div>

<h1>My <span class="note">Important</span>
Heading</h1>
<p>This is some <span
class="note">important</span> text.</p>

</body>
</html>
```

Figure 5. HTML example

Hello World

My **Important** Heading

This is some *important* text.

Figure 6. Web page example

Now that the structure of html files is clear, the method used to detect ads in these files will be explained.

The first thing that was done to address this problem, was to manually analyse a large number of different websites in search of patterns and structures that would help to identify the spaces reserved for ads, for this, the web page inspection tool offered by default by Google Chrome can be used and that allow to view the HTML code corresponding to each element of the web page. to access this tool, you must press the F12 key within the page you want to inspect.

Figure 7. Google Chrome inspection tool

What was discovered with this analysis was:

1. First, the vast majority of ads are grouped under the <div> element and in most cases have a class attribute associated with them. The name of the ad classes often contains words that can reveal whether or not it is an ad, as can be seen in the image below, where one of the ad classes is called "voc-advertising".

Figure 8. Advertising keyword with the inspection tool
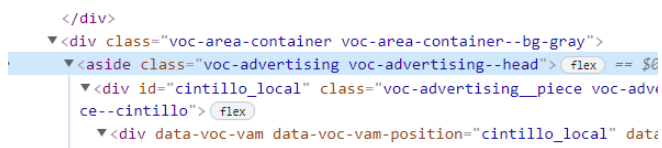

Figure 9. Advertising keyword in the HTML file

2. Secondly, many websites, especially the most visited sites, use the services of Google Ads. Google ads is an online advertising solution used by businesses to promote their products and services on Google Search, YouTube and other sites across the Web. These ads generated by Google, always have associated in their elements the class name "GoogleActiveViewElement".

Figure 10. GoogleActiveViewElement with inspector tool

Figure 11. GoogleActiveViewElement in the HTML file

3. Third and finally, it is quite impossible to go through all the websites in the world and study all the specific casuistry of each site. That's why the research was directed to sites where other ad blockers like AdBlock were detected so that by implementing the solution it could get the most out of both ad blockers simultaneously.

Figure 12. AdBlock detection

With these conclusions, the solution adopted was to hide those elements whose class name contained any word related to advertising or contained the class name of the Google Ad's elements: "GoogleActiveViewElement". To do this, a list was generated with the set of words that serve to identify an ad, hereafter referred to as <u>keywords</u>.

IV. IMPLEMENTATION

In this section the main elements that make up an extension and how the solution found in the previous section has been developed to search for and remove advertising in HTML files on websites is going to be explained.

Extensions are composed of 2 main elements:

- **Manifest**: the manifest is a JSON (JavaScript object notation) file, which defines the extension's metadata, such as the extension's author, description, JavaScript files used, etc.

- **JavaScript file:** this file contains the set of instructions that the browser must execute to fulfil the extension's objective.

In this particular case, the Manifest file, includes the name, description, version, author and manifest_version of the extension:

Name	AdBlock-HTML
Description	Master's thesis in Cybersecurity, consisting on the creation of an extension for the Google Chrome browser capable of blocking advertising on web pages, based on the detection of advertising in the HTML files of the pages and subsequent removal of the

	corresponding HTML code fragment
Version	0.1
Author	Diego García Garín
Manifest version	0.2

Table 1. Manifest attributes

The manifest also includes the permissions object, this object set the permissions that the extension will have, in this particular case:

Permissions	Description
Tabs	This permission allows the extension to access the tabs that the browser has opened.
webNavigation	This permission allows the extension to receive a notification about the status of the navigation requests on real time.
<all_urls>	This allows the extension to access the resources of whatever address that matches this expression. In this case, matcher any URL that starts with "http", "https", "file" or "ftp". So that almost all the web pages can be accessed by this extension.

Table 2. Extension permission

Finally, the last object of the manifest is the "background". In this section, the JavaScript that are going to be executed are listed, so that the browser knows the instructions to execute when the extension starts.

Regarding the scripts:

- **Extension.js:** this file is the first one that is executed. It executes in the background and it is always listening the notifications about the status of the navigation requests. When a new web page is completely loaded, then it checks if there are "iframes" also loading, if there isn't it executes the removeadds.js script.

- **Removeadds.**js: the first thing that it is included in this file is the <u>keywords</u> that will be used to check the class names. This script executes on an infinite loop that repeats each 200 milliseconds to ensure that all adds disappear in case new ads appear after the page has been loaded, e.g., if the user scrolls the page, script that generates ads, etc.

  To detect ads, first of all, it iterates through all the classes defined in the document and stores them in a local variable. Then each class name is compared to each name of the <u>keywords</u>, searching not only for exact coincidences, but also if the class name contains a keyword. This

has been done because there are some situations where a keyword is hide between other characters, for example, the class "advertising-number-one".

If there is a match, that class name is send to a function that removes that hides that element with the following command:

```
element.setAttribute("style", "display: none !important;");
```

V. RESULTS

In this section, the results obtained in the experiments carried out will be analysed.

Three experiments have been carried out to test the effectiveness of the created publication blocker. The main objective of these experiments is to test the effectiveness of the extension, both alone and in conjunction with other ad blockers on the market, in particular tests have been carried out in conjunction with AdBlock, the most widespread conventional ad blocker.

The method followed to carry out the experiments was to load a page with the extension deactivated, study the number of ads that appear, then activate the extension and record the number of ads blocked.

A. Experiment on known pages:

The first experiment consists of testing the extension on the web pages where the information has been extracted to generate the keywords.

Here is a table that shows the results of all the experiments:

Domain	Nº ads (initial)	Nº ads (extension)
Stackoverflow	2	1
ABC	33	2
ElMundo	10	0
As	10	0
Marca	12	2

Table 3. Experiment on known pages

Here are some examples of web pages without and with the extension:

Figure 13. Example 1 with ads

Figure 14. Example 1 without ads

Here are some examples of web pages without and with the extension:
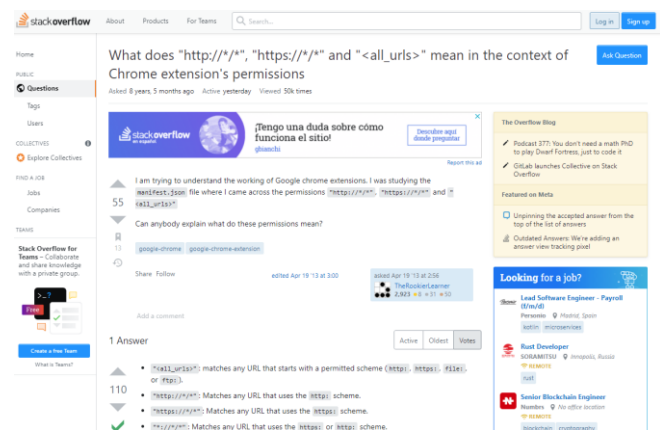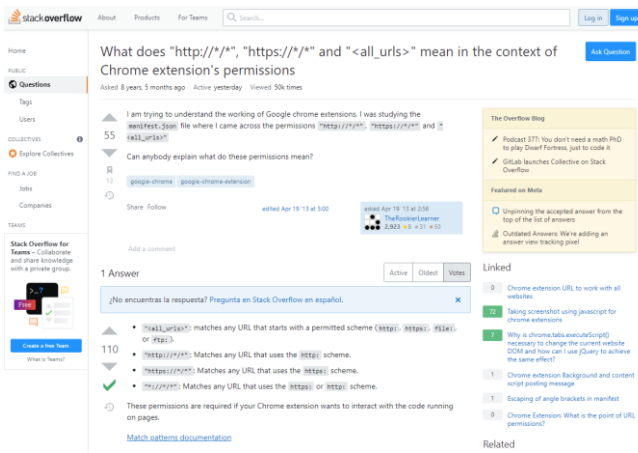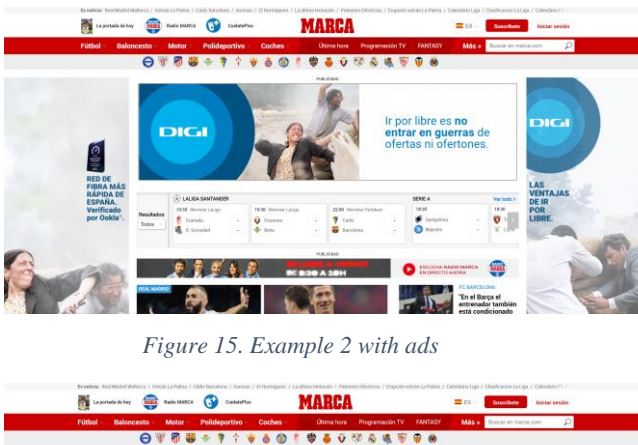

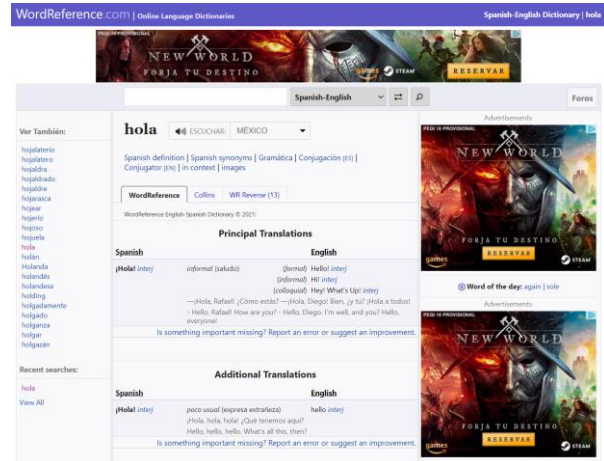*Figure 17. Example 3 with ads*

Figure 15. Example 2 with ads

Figure 18. Example 3 without ads

Figure 16. Example 2 without ads

B. Experiment on pages unknown by the extension but accessible by AdBlock:

The second experiment is on web pages where no information has been extracted but AdBlock is able to block.

Here is a table that shows the results of all the experiments:

Domain	Nº ads (initial)	Nº ads (extension)
Wordreference	3	0
Youtube	1	0
ElMundo	10	0
ElPais	7	0
ElDiarioNY	16	0

Table 4. Experiment on pages unknown by the extension but accessible by AdBlock
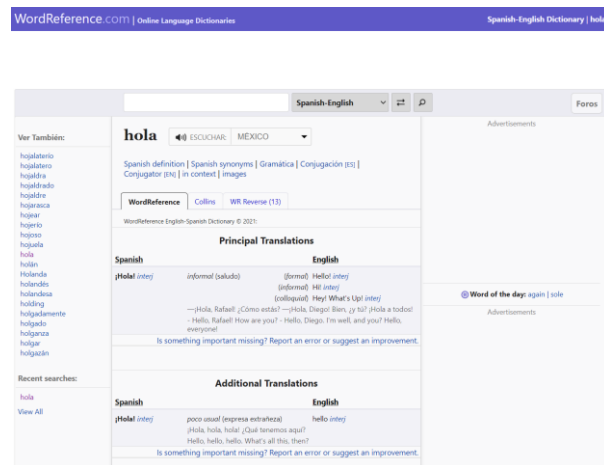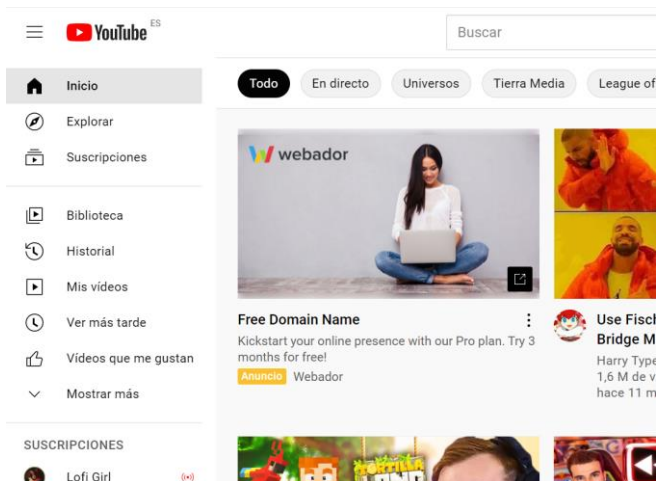

Figure 19. Example 4 with ads
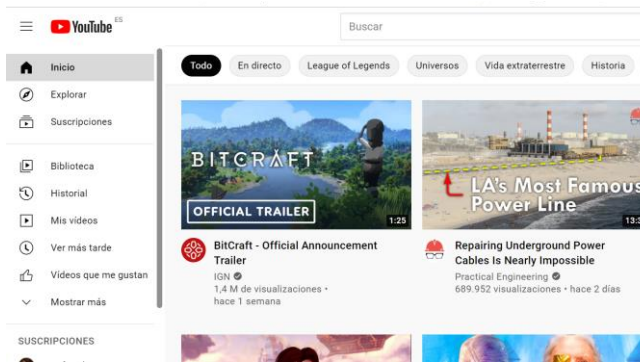
Figure 20. Example 4 without ads

Figure 22. Example 5 without ads

C. *Experiment with pages unknown to the extension and where AdBlock is detected:*

Finally, it has been tested on websites where AdBlock is not able to block advertising and have not been taken into account when generating keywords.

Here is a table that shows the results of all the experiments:

Domain	Nº ads (initial)	Nº ads (extension)
Wordreference	3	0
Youtube	1	0
ElMundo	10	0
ElPais	7	0
ElDiarioNY	16	0

Table 5. Experiment with pages unknown to the extension and where AdBlock is detected

Here is an example of a web page been without and with the extension:
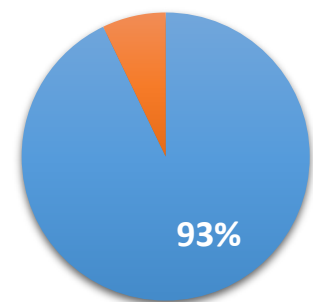

Figure 21. Example 5 with ads

D. *Average percentage of ads blocked:*

On average, after all experiments, an ad blocking percentage of one percent is achieved:

Total Number of Ads	Total Number of Ads Blocked	Percentage
156	12	92,86%

Table 6. Final results

Figure 23. Average percentage of ads blocked

VI. CONCLUSIONS

The main objective of this project was to create an extension for the Google Chrome browser capable of blocking advertising on different websites by using parsing techniques and modifying the HTML code of these sites.

To meet this objective, three sub-objectives were defined: the detection of advertising in HTML files, the implementation of the extension in the web browser and the analysis of the results.

The first sub-objective consisted of carrying out a study of the structure, characteristics and distribution of advertising on different websites, with the idea of identifying those factors that can be used to identify advertising on any website. This objective has been successfully achieved and all these concepts have been explained in the third chapter of this document.

The second sub-objective addressed the implementation of the extension in the Google Chrome web browser using HTML and CSS the layout languages that allows to define the structure and styles of a web page and the JavaScript programming language. This objective has also been successfully completed, and the implementation code has been included in Annex I of this document.

The third and final proposed objective was to perform a detailed analysis of the results of the implementation. This objective has also been successfully achieved by presenting the results obtained from the implementation in the fifth chapter of this document, where it is concluded that the implementation of the proposed extension has been satisfactory and that the results demonstrated in terms of ad blocking are good enough. Furthermore, it has been concluded that the implemented extension, when combined with the capabilities offered by conventional ad blockers, brings out its full potential, supporting each other and blocking almost all advertising on the Internet.

Therefore, it can be concluded that the proposed objective has been achieved and that an extension capable of blocking advertisements by modifying HTML files has been implemented.

As future steps, it would be interesting to work on implementing this extension in other browsers such as Firefox or Safari.

Even so, as long as people continue to consume content, advertising will continue to serve to subsidise it. But thanks to ad blockers and the growing community of users who rely on them, advertisers are finally starting to think carefully about the quality and number of ads they use to reach potential customers. There is already a consensus building in the industry that the only long-term solution is to regain users' trust and improve the quality of online ads, not to force ads on consumers who don't want to see them, and I hope that more projects like this one will come to light that can help reinforce this idea even further.

VII. REFERENCES

[1] Wikipedia, «Publicidad,» 13 08 2021. [En línea]. Available: https://es.wikipedia.org/wiki/Publicidad. [Último acceso: 01 09 2021].

[2] El Orden Mundial, «La población de Internet en el mundo y proporción de población con acceso a la red,» 2021. [En línea]. Available: https://elordenmundial.com/mapas/mapa-acceso-internet-mundo/. [Último acceso: 01 09 2021].

[3] Statista, «¿Cuántas horas al día pasamos conectados a Internet?,» 28 01 2021. [En línea]. Available: https://es.statista.com/grafico/22701/tiempo-medio-de-uso-diario-de-internet/. [Último acceso: 01 09 2021].

[4] AdBlock, «A How To Guide to Ad Blocking,» 06 03 2017. [En línea]. Available: https://blog.getadblock.com/a-how-to-guide-to-ad-blocking-be452ed5ed6f. [Último acceso: 01 09 2021].

[5] Expansion, «¿Cómo cautivar al consumidor que recibe 5,000 impactos publicitarios al día?,» 28 01 2020. [En línea]. Available: https://expansion.mx/mercadotecnia/2020/01/28/como-cautivar-consumidor-que-recibe-5000-impactos-al-dia. [Último acceso: 01 09 2021].

[6] Statista, «Market share held by leading desktop internet browsers in the United States from January 2015 to June 2021,» 20 07 2021. [En línea]. Available: https://www.statista.com/statistics/272697/market-share-desktop-internet-browser-usa/. [Último acceso: 01 09 2021].

[7] B. Dean, «Ad Blocker Usage and Demographic Statistics in 2021,» Backlinko, 09 03 2021. [En línea]. Available: https://backlinko.com/ad-blockers-users#ad-blocker-usage-worldwide. [Último acceso: 01 09 2021].

[8] AdBlock, «How does AdBlock work?,» 27 08 2020. [En línea]. Available: https://help.getadblock.com/support/solutions/articles/6000087914-how-does-adblock-work-. [Último acceso: 01 09 2021].

[9] AdBlock Plus, «Easylist,» 22 09 2021. [En línea]. [Último acceso: 22 09 2021].

[10] Detect AdBlock, «Most effective way to detect ad blockers,» 2021. [En línea]. Available: https://www.detectadblock.com/. [Último acceso: 01 09 2021].

[11] MDN Web Docs, «HTML: Lenguaje de etiquetas de hipertexto,» 21 09 2021. [En línea]. Available: https://developer.mozilla.org/es/docs/Web/HTML. [Último acceso: 21 09 2021].

[12] W3Schools, «HTML Tutorial,» 2021. [En línea]. Available: https://www.w3schools.com/html/. [Último acceso: 01 09 2021].

[13] Google Inc., «Chrome Developers,» 28 February 2014. [En línea]. Available: https://developer.chrome.com/docs/extensions/mv3/getstarted/. [Último acceso: 01 September 2021].

ANNEX I

The main code files that have been developed can be found in this annex:

Manifest.json:

```json
{
    "name": "Addblock-HTML",
    "description": "Master's thesis in Cybersecurity, consisting of the creation of an exten
sion for the Google Chrome browser capable of blocking advertising on web pages, based on th
e detection of advertising in the HTML files of the pages and subsequent removal of the corr
esponding HTML code fragment.",
    "version": "0.1",
    "author": "Diego García",
    "manifest_version": 2,
    "permissions": [
        "tabs",
        "webNavigation",
        "<all_urls>"
    ],
    "browser_action": {
        "default_title": "Addblock-HTML"
    },
    "background": {
        "scripts": [
            "extension.js",
            "removeads.js"
        ]
    }
}
```

Extension.js:

```javascript
/*
    This extension starts when the browser has completed loading a webpage, to ensure that
    we remove all the ads, and make sure that we dont remove legitimate content of the page.
    Another option is to use webNavigation.onCommitted, this will run when the browser has
    committed to loading a webpage, this will start to run early so that we can begin to rem
ove
    ads as soon as possible, however, this may result in legitimate parts of the website's
    content being deleted.
*/
chrome.webNavigation.onCompleted.addListener(function (tab) {
    // Prevents script from running when other frames load, to be able to remove the ads on
those frames
    if (tab.frameId == 0) {
        chrome.tabs.query({ active: true, lastFocusedWindow: true }, tabs => {
            try {
                runLinkedinScript();
                return;
            } catch (err) {
                throw err;
            }
```

```
        });
    }
});


function runLinkedinScript() {
    // Inject the script to remove ads from file into the webpage
    chrome.tabs.executeScript({
        file: 'removeads.js'
    });
    return true;
}
```

Removeads.js:

```
// List of key words to remove
var adv_list = ["adv","ads","GoogleActiveViewElement","advertisement","google_ad","googleads
", "-ad-", "advertising", "publi", "publicidad", "-web-ad-"];

// Deletes all the specified classes from the html document
function detectAd(ad_key_word) {

    let by_class = document.getElementsByClassName(ad_key_word);
    let by_id = document.getElementById(ad_key_word);

    let spans = by_class;
    console.log(spans);

    if (spans.length == 0) {
        if (by_id != null){
            removeAd(by_id);
        }
    }

    for (let i = 0; i < spans.length; ++i) {
        let ad = spans[i];
        removeAd(ad);
    }
}

// Hides the specified element
function hideAd(element) {
    element.setAttribute("style", "display: none !important;");
    console.log(element + " --> has been hided!");
}

// Removes the specified element
function removeAd(element) {
    element.remove();
```

```javascript
        console.log(element + " --> has been removed!");
}

// Detects all the classes that might haver ads on them
function detectAdKeyWords(){
    var allClasses = [];
    var allAds = [];

    var allElements = document.querySelectorAll('*');

    for (var i = 0; i < allElements.length; i++) {
    var classes = allElements[i].className.toString().split(/\s+/);
    for (var j = 0; j < classes.length; j++) {
        var cls = classes[j];
        if (cls && allClasses.indexOf(cls) === -1)
        allClasses.push(cls);
        }
    }

    for (var x = 0; x < allClasses.length; x++) {
        let str = allClasses[x];
        for (var y = 0; y < adv_list.length; y++){
            if (str.includes(adv_list[y]) || str.localeCompare(adv_list[y]) == 0){
                allAds.push(allClasses[x]);
                detectAd(allClasses[x]);
            }
        }
    }
    console.log(allAds);
}

detectAdKeyWords();

/*
    Ensures ads will be removed periodically in case new ads appear after the page has been
loaded,
    e.g., if the user scrolls the page, script that generates ads, etc.
*/
setInterval(function () {
    detectAdKeyWords();
}, 200)
```

ANNEX II

Here is the list of all the web sites visited to test de extension:

- https://elpais.com/

- https://www.20minutos.es/noticia/4830174/0/despertadores-digitales-por-que-es-malo-usar-la-alarma-del-movil-y-tres-buenos-modelos-para-evitarlo/?autoref=true

- https://www.elcorreo.com/butaca/en-rodaje/joaquin-phoenix-jodie-20210916191944-ntrc.html

- https://www.diariosur.es/sociedad/quedaria-ciudad-cayera-20180605133018-nt.html?ref=https:%2F%2Fwww.google.com%2F

- https://www.abc.es/

- https://www.elmundo.es/

- https://as.com/

- https://www.marca.com/

- https://expansion.mx/mercadotecnia/2020/01/28/como-cautivar-consumidor-que-recibe-5000-impactos-al-dia

- https://stackoverflow.com/questions/16096482/what-does-http-https-and-all-urls-mean-in-the-context-of-ch

- https://www.lavanguardia.com/vida

- https://eldiariony.com/2016/11/26/el-video-de-la-cabra-que-arrasa-en-whatsapp-no-es-lo-que-parece/

- https://okdiario.com/curiosidades/como-cocodrilo-del-nilo-4904949

- https://www.elespanol.com/ciencia/nutricion/20210923/superlegumbre-proteinas-huevo-fosforo-salmon/613939911_0.html

- https://www.medicalnewstoday.com/articles/es/ampolla-en-el-labio#diagnostico

- https://www.lavanguardia.com/tecnologia/internet/20210811/7656083/vuelve-the-pirate-bay-pmv.html

- https://as.com/diarioas/2021/04/04/actualidad/1617522301_249725.html

- https://www.trendencias.com/belleza/calendarios-adviento-belleza-maquillaje-2021-2022-deseados-cuando-salen-a-venta-donde-puedes-comprarlos

- https://www.elcorreo.com/butaca/en-rodaje/joaquin-phoenix-jodie-20210916191944-ntrc.html

- https://elpais.com/escaparate/2021-09-23/como-limpiar-la-plancha-de-la-ropa-y-productos-utiles-para-ello.html

- https://www.levante-emv.com/sociedad/2021/09/23/perseverance-marte-roca-apunta-existencia-vida-57477114.html

- https://www.20minutos.es/noticia/4830174/0/despertadores-digitales-por-que-es-malo-usar-la-alarma-del-movil-y-tres-buenos-modelos-para-evitarlo/

- https://www.abc.es/motor/reportajes/abci-artefacto-peregrino-santiago-compostela-202109230117_noticia.html

- https://www.elespanol.com/ciencia/nutricion/20210923/superlegumbre-proteinas-huevo-fosforo-salmon/613939911_0.html

- https://www.youtube.com/

- https://www.wordreference.com/es/en/translation.asp?spen=hola

- https://www.efe.com/efe/espana/1

- https://cincodias.elpais.com/

- https://www.elconfidencial.com/

- https://www.elcorreo.com/

- https://www.elperiodico.com/es/

- https://www.libertaddigital.com/

- https://www.larazon.es/

- https://thenewsonline.es/

- https://www.bbc.com/news/science-environment-33517492

- https://librosaguilar.com/dontorrent/

- https://www.lainformacion.com/empresas/movistar-vodafone-y-orange-encabezan-las-perdidas-de-las-lineas-moviles/2849428/