CSAI

COC257

F122151

**PROJECT TITLE – Search Engine**

*by*

*Jonah Wasilewski*

*Supervisor: Dr P. Norris*

*Department of Computer Science*

*Loughborough University*

May/June 2024

# Abstract

Table of Contents

[Abstract 2](#_Toc155717863)

[1 Introduction 5](#_Toc155717864)

[1.1 Goals 5](#_Toc155717865)

[1.2 Why Undertake the Project 5](#_Toc155717866)

[2 Literature Review 6](#_Toc155717867)

[2.1 History of Web Browsers 6](#_Toc155717868)

[2.2 Web Crawling 7](#_Toc155717869)

[2.3 Search Engine 8](#_Toc155717870)

[2.4 Mobile Web Browsers 9](#_Toc155717871)

[2.4 Accommodating Designs 9](#_Toc155717872)

[2.5 Cookies 11](#_Toc155717873)

[2.6 Web Browser Security 12](#_Toc155717874)

[2.6 AI Methodologies 13](#_Toc155717875)

[2.7 AI Uses in Web Browsers 15](#_Toc155717876)

[2.8 Development Process Models 16](#_Toc155717877)

[2.8.1 Waterfall 16](#_Toc155717878)

[2.8.2 Spiral 16](#_Toc155717879)

[2.8.3 V Model 17](#_Toc155717880)

[2.8.4 Incremental 17](#_Toc155717881)

[2.8.5 Agile 17](#_Toc155717882)

[3 Abbreviations and Acronyms 18](#_Toc155717883)

[4 Additional Transferable Skills 18](#_Toc155717884)

[5 Project Journey??? 19](#_Toc155717885)

[6 Design 20](#_Toc155717886)

[6.1 Overview 20](#_Toc155717887)

[6.2 Privacy Settings 21](#_Toc155717888)

[6.3 Search History 21](#_Toc155717889)

[6.4 Database Designs 21](#_Toc155717890)

[6.4.1 Users 21](#_Toc155717891)

[6.4.2 Websites 22](#_Toc155717892)

[6.4.3 Search History 23](#_Toc155717893)

[6.5 Classes 24](#_Toc155717894)

[6.5.1 Website 24](#_Toc155717895)

[7 Implementation 25](#_Toc155717896)

[7.1 Web Crawler 25](#_Toc155717897)

[7.2 Search Engine 25](#_Toc155717898)

[7.2.1 Register 26](#_Toc155717899)

[7.2.2 Sign In 26](#_Toc155717900)

[7.2.3 Log Out 26](#_Toc155717901)

[7.3 AI Data Extraction 26](#_Toc155717902)

[7.3.1 Cookies 26](#_Toc155717903)

[7.4 AI Data Processing 27](#_Toc155717904)

[7.5 AI Data Uses 27](#_Toc155717905)

[8 Results 28](#_Toc155717906)

[9 Interpretation of Results 29](#_Toc155717907)

[10 Figures and Tables 30](#_Toc155717908)

[11 Conclusion 31](#_Toc155717909)

[12 Future Work 32](#_Toc155717910)

[13 References 33](#_Toc155717911)

# 1 Introduction

## Goals

The ultimate goal of the project is to build a search engine that can be a powerful and user-friendly tool for anyone to retrieve information on the web. My project will consist of several key elements, starting with the design and implementation of a web crawler. The task of the web crawler is traversing the internet, indexing web pages, and collecting data to make it searchable. This first phase is very important as it will be used to create a ‘crawl list’ that the web program will use to search for relevant webpages based on the user’s search query.

The user interface (UI) is a relatively simple section of the program but is also an integral aspect because it needs to be easily understandable and useable in order to ensure that the user has the best possible experience while interacting with the program. A well-designed UI should be aesthetically pleasing and make it easy for users to enter their queries and navigate through search results efficiently. This includes the use of website summaries that give the user a general overview of the website before they visit the page itself so that they can make the best use of their time while looking for a piece of information.

The core functionality of the search engine is to provide relevant search results based on a user’s search query. This involves using algorithms for indexing, ranking, and retrieving web pages from a crawl list. The speed at which the search can provide these results is of the highest importance because no user wants to be sat waiting a long time for the program to return their results, and so it’s not solely about how good the results are but also the speed they can be retrieved.

Incorporating artificial intelligence (AI) elements like natural language processing (NLP) and user profiling is a way to take the project to the next level and separate it from many other search engines. NLP can be used to extract key data from webpages to create summaries, giving users concise and easily digestible information. User profiling enables a personalised experience by tailoring search results to individual preferences and behaviour, enhancing the user experience further. This is done by looking at recent searches, activity and interaction to analyse the user’s interests and trends.

The end goal of this search engine project is to provide users with an inventive and highly customizable search experience. It aims to empower users to find the information they need on the in a way that sets it apart from generic search engines with this intelligent way to find information on the internet.

## 1.2 Why Undertake the Project

AI has quickly become one of the most influential and useful tools for a wide range of professions, especially in computer science. The typical and most commonly thought of type of AI in recent times is the AI chat bot, with ChatGPT being the frontrunner with STATS FOR CHATGPT AND OTHER AI CAHT BOTS. This tool brings a wide range of knowledge from across the Internet into one, easily accessible place where a user has the ability to research, learn and test their understanding of any topic they want. An obvious extension of this technology is incorporating it with a web browser. Many well trusted and well established search engines already exist and so an infrastructure that gives any person with Internet access the ability to FINSIH is already in place With this being said, it can often be hard

# 2 Literature Review

Due to the importance of web crawling and search engines, there is a good amount of literature and research available that gives detailed explanations on one or multiple methods for each of these topics. Since the first PageRank algorithm was proposed by Larry Page and Sergey Brin, the founders of Google, in 1988 (Yu, Li and Zeng, 2021), computer scientists have worked on ways to improve and refine the method in which their algorithm goes about following the links contained in webpages and using these to determine the most relevant and useful webpages to a user. Hence, there are multitudes of these improvements documented but it is important to know that the context in which a specific algorithm is used can, and mostly likely will, determine how it works.

The purpose of this literature review is to explain the methodologies and techniques developed and used by software developers over many decades in order to reach the level of understanding and complexity that we see in the web browsers we use in our everyday lives. As well as this, the review will form the basis of explaining my reasoning and the logic behind the choices of implementation strategies used throughout my project.

## 2.1 History of Web Browsers

The Web was first invented in 1989 by Tim Berners-Lee. It was initially designed to provide for the demand for information-sharing among institutions and universities around the world and as a result this model only offered very basic functionality. Only reading data was possible because there was no way for a person to post themselves or interact with the content. The technologies used were HTML, HTTP, URI and JavaScript, along with protocols like XML, XHTML and CSS. Overall, the Web 1.0 was extremely slow and lacked a lot of functionality due to it only working in one direction.

In 1999, Darci DiNucci released the Web 2.0 (also called The Participative and Social Web) but it didn’t become widely used until 2004 and it is still the current version of the World Wide Web used today. It allowed a user to post content and, as an extension, connect with other people also using this new Web (meaning it was bi-directional, a significant difference compared to the Web 1.0) which was revolutionary and created a wide range of new possibilities. The introduction of technologies such as XML, DOM and REST allows a creator to include interactivity and communications with a client. However, with the introduction of these new functionalities, insecurity also crept into the infrastructure of the Web, specifically, they allowed a user to be hacked because a person can add their own, potentially malicious, programs that can be a threat to another user. Therefore it has become essential for developers to think carefully about how they can ensure user safety when interacting with their website. (Shakir M. Abass, 2019)

Web 3.0 is a theoretical model that researchers believe is the next iteration of the Web. It is a set of values and applications that define the new age of the World Wide Web, but it is only a prediction on what people expect the current version to evolve into in the future and therefore it is hard to say for sure what the final product will look like. There is evidence of Web3 in use today, for example the Metaverse is an example of an immersive web that allows a user to interact even closer to the information that they want to find as well as other people on the Internet. Firefox now supports WebVR and A-Frame, which let developers build VR websites in faster times than ever, and this will only speed up the process of converting from a Hyper Text based web to an immersive one. The blockchain is a decentralized, distributed ledger that is used to record transactions with the help of a

large collection of computers ensuring that the transaction can’t be altered or incorrectly recorded because all computers keep and record and can say when there has been an error. It is said that Web3 will be built on top of blockchain technologies so that this same principle of decentralization and error free transactions can be applied to create a faster, more secure platform. (*What Is Web 3.0 (Web3 definition)?*, no date)

It is no surprise that the first graphical web browser came at the same time as Tim Berners-Lee released Web 1.0 to the public because a person using the web would need a way to search through all the data stored to find what they were looking for. He called this access point to the internet the WorldWideWeb and a year later, a Line Mode browser was created to allow internet access through basic computer terminals.

As the popularity of this browser grew, companies saw a big opportunity and began thinking of ways in which they could benefit from it. In 1993, Mosaic was created at the NCSA, which came exceedingly popular and is the early ancestor of Mozilla Firefox. Then, in 1994, Andreessen released Netscape Navigator, which was the first step towards a so called ‘Browser War’. Following this in 1995, Windows licensed the old Mosaic code and released its own Internet Explorer and from then on Netscape and Microsoft competed with each other to create the best browser.

A big milestone that came out of this war was JavaScript, invented and released by Netscape, which gave websites lots more computing capabilities and functionalities. Shortly after, Microsoft was responsible for the release of CSS that helped style webpages to make them easier to use and gave them a nicer overall appearance. But when Microsoft started to equip all their Windows operating systems with Internet Explorer, their dominance was quickly established and by 1999 they controlled 99% of the market. In 2004 Netscape open sourced its codebase to create Mozilla, a non-profit web browser that gave users a choice. By 2010, Internet Explorer’s market share had dropped down to 50%. Other competitors arose around this time as well, such as Safari and Google Chrome, further reducing the popularity of Internet Explorer. In 2015, Windows 10 saw the replacement of Internet Explorer with Microsoft Edge.

(*Browser History: Epic power struggles that brought us modern browsers*, no date)

## 2.2 Web Crawling

Web Crawling has been widely used for decades now due to the need to extract copious amounts of data from the Internet. Due to the nature of the format of data, namely that it is unstructured, it can be a challenge to take away the valuable information that is needed for a specific purpose. Web scraping has many valuable uses in a wide range of areas such as: a price comparison site looking at different company’s offers for a quote, analysis of social media data, or making accurate stock market predictions. But in the context of a web browser, a web crawler simply has to follow a series of websites by looking at the hyperlinks that are present in each one so that it can record websites that it finds on its way.

The challenge is creating a foolproof method that allows the crawler to handle websites where they will have to enter a username and password. It is not possible to code a crawler that has a valid account for any site that it may come across and it is equally as impossible for the crawler to have the ability to guess the correct details to get passed this security. And so a crawler is usually given input directives that allow it to “assign specific values to specific input fields” (Liu, Chen and Sun, 2020) so that the crawler can apply these input directives in fields as needed. A number of issues arise through the use of input directives. Namely, it is very difficult for a developer to know what the format of the fields will be like in an unknown website and so it is incredibly difficult to design the crawler in a way that will be able to deal with an unknown format, and it will be necessary to analyse the text in a webpage in order to figure out the required inputs which is a tedious and time consuming process.

## 2.3 Search Engine

For the most part, all examples of modern search engines that you can find on the internet (e.g. Google, Bing, etc) all follow the same format and principles. There will always be the name of the search engines and the search bar as the main two features of the page. These are the essentials for a webpage GIVE AN EXAMPLE OF A SOURCE THAT SUPPORTS THIS. Additionally, it is common to see the websites that the user most frequently visits so that they can have quick and easy access to then without the need to type them in the search bar. Following on from this concept of remembering search history, when a user starts to type a query into the search bar, lots of existing web browsers try to predict what they want to search and will autofill or give personalised suggestions to try and speed up the search process.

It is also common practice for a web browser to offer a settings page where the user can set any preferences that they would like. For example, switching between light and dark mode (but this may also be offered by the application or operating system that they are using as well), or whether the user is ok with the storing of their relevant search data. Additionally, a user may want to delete their search history, they may want to remove all of it or just a specific page or time interval and so it is very common to see an option to facilitate this in the settings page of a web browser. A more advanced feature that can be offered to the user is the ability to turn on safe search where the user is protected from any malicious data or programs on websites that can cause harm to their device. Protection like this can be achieved by using a blacklist to mark pages as dangerous, or by scanning the content of a website to find any potential threats that the user should be warned about and blocked from accessing (Hr *et al.*, 2020). An obvious extension of this is to create a child-friendly search environment in which a parent or guardian can be sure that their child won’t be able to access sites with dangerous or inappropriate content. Another choice that can be offered by the browser is the ability to enable or disable JavaScript in order to offer protection against security threats and malicious attacks. However, doing this comes at the expense of the user experience and so should be offered to the user only in situations where the page wont greatly suffer. For example, if a page relied on JavaScript for animations and other interactive features then disabling these will reduce the helpfulness and readability (Das, 2022). Sometimes a website will want to use a user’s location to help with the service they provide, for example in a weather forecast or finding the closest shop to the device’s current location). This can obviously be very helpful but some people might not feel comfortable with being tracked and having their location stored, therefore it is good to have an added feature that lets the user block any requests or attempt to locate their whereabouts. The same is true for other request like using the device’s camera that a user might feel uncomfortable about.

Another very common practice in a search engine is the use of pagination. This is the process of splitting up the list of websites that have been shown to be some sort of match to the user’s query into smaller groups and only displaying one group at a time. These groups are also called pages. On the webpage, the user is shown a sequence of numbers from one to the number of pages and has the option to click on any of these numbers to bring up the websites belonging to that page. This allows the user to see only a small number of webpages at a time so that they don’t have too much data to look through at one time, whilst also providing insight into the relevance of a website by the page that it appears in. The best example of this feature in a search engine is Google’s implementation of pagination. If you were to go to the bottom of the search results page after entering a query, you will see Google written with lots of o’s and these o’s are links to a specific page returned by the search engine.

## 2.4 Mobile Web Browsers

Making any webpage compatible with a mobile device is a necessity for any web developer these days, and a search engine is no exception. Due to the rise of mobile devices, starting from the release of the first iPhone in 2007, mobile web browsing technologies have boomed in a very short span of time and this has led to a number of competing versions to emerge. Among these are Google Chrome and Safari. “Many information technology companies have recognized the usage share of their mobile Web browsers as an important marketing strategy” (Ahn, Kim and Proctor, 2018) and therefore have made a significant effort to create the best UI for their browser. The many task that these companies have had to deal with is working with the greatly reduced screen size that comes with using a mobile device compared to a larger one (like a desktop computer), and so it is important to try and maximise usability, learnability and usefulness. (Ahn, Kim and Proctor, 2018)

It should be noted that although this is an important aspect of a web browser due to the magnitude of mobile devices in the world today, the main focus of this project is to implement additional machine learning and AI elements to the program and so I will not focus on it too much when it comes to the implementation stage.

## 2.4 Accommodating Designs

Another highly relevant area for website design in the modern world is incorporating an accommodating design to allow for easy access to any person with visual, audio or any other kind of impairment. Even though it does again fall outside of the scope of my project, it is good to acknowledge the extra design considerations that would go into a web browser that would be released for public use.

Increasing web accessibility for older adults (Sa-nga-ngam and Kiattisin, 2020). Study aimed to create a redesigned prototype web browser that allowed for, and can be proved to be, more user friendly for the older demographic of people (people over 65 years old which is around 10% of the world’s population). As of 2018, 90% of the world used the Internet but on 78% of people aged 65 and over were reported to actively use the Internet. It is this divide in generations that has come to be a concern and a problem that [Prush Sa-nga-ngam](https://link.springer.com/article/10.1007/s11277-020-07220-6#auth-Prush-Sa_nga_ngam-Aff1) and [Supaporn Kiattisin](https://link.springer.com/article/10.1007/s11277-020-07220-6#auth-Supaporn-Kiattisin-Aff1) sought to analyse and find a solution for in order to provide support to older adults living in this new digital age. A few of the main findings of the study show that creating a more simplistic UI, adding assistive features (stopping ads, removing animations, hiding pictures, etc) and creating a new layout design were the most effective ways in which to improve user experience.

HCI and culture (Kyriakoullis and Zaphiris, 2016). A big challenge in the design of a user interface is to be accommodating to the diverse cultures that will potentially interact with the system. It is essential to understand cultural values to create a successful and widely accepted system, but this can often be an almost impossible task as there is so much that a designer and developer will have to consider. In fact, “research community examines those cultural characteristics that may partially be responsible for the slow uptake or rejection of a system” (Kyriakoullis and Zaphiris, 2016). The study found that the most important consideration HCI is the user interface because it is the first impression that the user gets of the website or system and so it can be as simple as the user rejecting it in the first few seconds of use. Marcus and Gould, for instance, report that it is very important to have a number of options to choose from in a high uncertainty avoidance culture in order to prevent users from getting lost in the system (Marcus and Gould, 2000). Additionally, it is particularly important to think about “invisible cultural attributes”, specifically the cultural attributes that affect a person’s or a society’s behaviour. In a study by Su and Adams the difference between two e-commerce sites, Amazon tailoring to an international market and Dangdang tailoring specifically to the Chinese population, and how it is very hard to apply the Amazon business model to a Chinese market (Su and Adams, 2005). For example, it has been found that while the typical British culture is to consult family and friends before making a purchase, while the complete opposite is true in Chinese culture and an individual would rather go through with a purchase by themselves. This, among many other characteristics, show that there is a completely different view online shopping depending on the culture and therefore extra attention is needed when trying to expand or accommodate to a different culture.

People with cognitive difficulties will typically face a number of challenges when it comes to interacting with and using a piece of software and it can help to analyse the reasons for these challenges in order to design a system that allows a person to get the most out of it(Gregor and Dickinson, 2007). As opposed to designing a system with people with physical impairments in mind where we are helping with dexterity problems, designing the system for people with cognitive impairments requires the focus to be on how we can help those who most likely don’t have the technical knowledge or skills to use the assistive features provided with many websites today. In a study, Dickinson et al. examined the useability of Microsoft Outlook Express and found that 50% of a group of older user’s failed to complete a list of basic email tasks, and it is this complexity that needs to be reduced in order to make a system more useable, especially when considering those with cognitive impairments.

## 2.5 Cookies

The GDPR says that the use of cookies to gather and store data to potentially create an identifiable profile of a user means that they qualify as personal data and therefore they are subject to the GDPR. As a result, this means that cookies should be stored as securely as any other piece of data stored by a company and said company is responsible for any breaches of this data (*Cookies, the GDPR, and the ePrivacy Directive*, 2019). Further legislation on cookies passed in 2002 and amended in 2009 in the ePrivacy Directive. Its most notable consequence was the proliferation of cookie consent popups. Its main purpose is to enhance the GDPR, which only contains one reference to cookies in Recital 30 (*Recital 30 - Online identifiers for profiling and identification*, 2018), in order to expand on the confidentiality of electronic communications and tracking internet users.

When cookies were first used on the Internet in 1994, they were designed to support added functionalities and improve the user experience on a webpage by remembering small details about interactions had by the user. But as time went on, companies realised the potential that cookies offered in terms of the collection and manipulation of user data and companies unlawfully processing this data has become quite a serious fear for many people. In the study by Alharbi et al., out of the chosen 200 websites, only 30% provided information to the user about what a third party does with the cookies that it extracts (Alharbi *et al.*, 2023). This is where the concern comes from for most people because there is not enough transparency and openness to the consumer about what data is being collected, how it is being user and the potential risks that are associated with the processing of said data. (Kumar and Sharma, 2015). There are regulations in place that are designed to enforce any provider or developer to be honest and abide by the ‘privacy by design’ principle rather than making the user do the work to keep their data safe. And so, a developer must now get the explicit request of the user in order to use coolies for the storing and processing of any data whatsoever.

The most acceptable practice when it comes to getting permission to the use of cookies is by allowing the user to accept or decline the collection of certain or all cookies. Sometimes cookies are labelled as ‘essential’ which means that they are needed for the general performance of the page and the website won’t be useable without them, but this does not remove the need for a developer to get consent from the user before tracking them (*How do we comply with the cookie rules?*, 2023). Others are ‘unessential’, which means that they are used to store additional information about the user that are used to create a user profile for first or third parties. Unessential cookies should be able to be disabled by the user at any point in time if they decide that they don’t want their data to be collected and this has to be complied with due to the ePrivacy Law. Third parties will use cookies for advertising to give the user relevant and targeted ads to pique the interest of a person into viewing a product or site. Doing so is well within the law, but a company must be careful with the ways in which the acquire the cookies. This means that they must be completely transparent with all of the ways that the cookies will be used and who will have access to them, and there should be no insecure handling of any user data. One famous example of where a company did not comply with cookie regulations was Google in 2021, who was heavily fined by the French data protection authority[Commission, Nationale de l’Informatique et des Liberté](https://www.cnil.fr/en/cookies-cnil-fines-google-total-150-million-euros-and-facebook-60-million-euros-non-compliance) **(CNIL), for the illegal use of cookies. On the French version of Google, the platform YouTube asked users to consent to the use of cookies in a way that was a lot easier to accept cookies than to reject them, this goes against the principle of ‘privacy by design’ and so Google was found guilty of the charges. There was a comparable situation with Facebook at the same time in which they were fined** €60 million and given three months to comply to the charges and change their system so that they fit legislation.

## 2.6 Web Browser Security

Blacklists (Virvilis *et al.*, 2015). Blacklists are a common tool that developers will use to mark a website as ‘rogue’ or untrustworthy. The purpose is to alert a web browser of a website that will likely cause harm to a user if they were to access it and therefore stop anyone from falling victim to the malicious program. There are many URL blacklists that can be used by developers, the most common being Google’s Safe Browsing and Microsoft’s SmartScreen, and these usually offer enough protection against common and traceable threats. The study from Virvilis et al. investigates the potential problems and shortcomings of the use of URL blacklists, and specifically finds that the effectiveness in which many anti-phishing tools find and block malicious sites was not up to the standard that it should be.

Phishing is a social engineering technique in which a hacker will pretend to be someone they’re not or create a fake scenario in order to steal important information from a victim (Mazher, Ashraf and Altaf, 2013). As previously mentioned, using a blacklist to alert a browser of any malicious websites isn’t enough to create a completely safe search environment and so extra protection is needed. This is where the heuristic based approach can be put to use. In this technique, a programmer can try to determine if a site is malicious by checking for one of many suspicious signs. This is done by checking the URL authenticity and scanning the content of the webpage. Machine learning algorithms have been introduced to perform these tasks which has led to the speed at which a malicious site is found to increase.

The most common and well known technique in improving the security of a web browser is the use of security toolbars. The provider of the search engine will often provide lots of different features that can help a user navigate through the internet and avoid any potentially harmful data or content. For example, if a user tried to download something that has been flagged or appears suspicious to algorithm, then an error message will popup asking the user if they are confident that what they are about to download is safe, or the browser may block the process altogether until the administrator of the system changes some permissions. A study performed by Mazher, Ashraf and Altaf investigated the effectiveness of how well a number of web browser performed in alerting a user of a potential threat on a website they are about to access. The browsers involved in the experiment were Internet Explorer, Firefox and Google Chrome. The results showed that out of the three browsers, only Google Chrome was successful in alerting the user to the threat that was posed by the phishing website they were trying to access. Even though this test was performed in 2013, the main principle of the test remains relevant to this day because it will always be important for a successful and safe web browser to detect threats to a user and alert them of it before it’s too late. Furthermore, the results showed that for Internet Explorer and Firefox, all of the users proceeded to use the phishing website because they believed it to be the real version of Facebook, and because neither browser alerted them to any kind of threat, they entered all of their account information, opening up the possibility of them being hacked. This further emphasizes the importance of a web browser providing state of the art protection for any user.

 A Quick Response (QR) code are used to represent data, like the address or URL for a website in a scannable form ready to be parsed by a consumer’s mobile device. This technology has allowed providers to make their platform more easily available by putting a code in a heavily visited space where it can draw attention. However, hackers have seen QR codes as an opportunity to launch a new kind of attack on a victim’s device. Commonly, if a consumer is told to scan a QR code in a leaflet for some produce then they will, even just out of curiosity, which gives hackers the opportunity to place links to malicious sites or data that will exploit the device of the scanner when accessed. Similarly, a hacker can also replace an existing QR code (for example, a code that has been pinned to a noticeboard) with their own, tricking any person who sees the QR code into thinking that it is a legitimate link when it is not. This type of cyber-attack is called QRishing (Vidas *et al.*, 2013). The study by Vidas et al. confirmed that the majority if people will scan a QR code out of curiosity rather than because they were interested in the relate information that the link contained, with 64% scanning for this reason. An additional 14% scanned the code for fun, highlighting the fact that the majority of people don’t consider the threat of hacking when scanning a QR code.

Another important consideration that a developer needs to make when it comes to website design in the secure transfer of data from the client to the server in order to store, process or retrieve information. There is the simple use of POST and GET in an HTML form that allows the programmer to specify the method in which the data is sent over to the server or another webpage and can be useful in the simple cases. The GET method sends the user information appended to the page request separated by a ‘?’, whereas the POST method transfers information via HTTP headers so that the information is encoded and put into a header called ‘QUERY\_STRING’. Hence, the POST method is more secure than the GET method and is the only option for sending sensitive user information from an HTML form to a server to be processed.

However, there are a number of mechanisms and software that can be used to further increase the security of internet transactions. A key example for the use of a web browser is PDO. This is an extension that makes data stored in a database safer from any SQL injection attack made by a hacker trying to gain access to the system or data inside it by entering SQL commands into the input boxes of a website. This is a very simple yet effective approach to trying to access data without the permission of the provider. PDO can be an added barrier to protect against such attacks because a programmer can make use of prepared statements in their code for storing or manipulating data in a database so that it is harder for a hacker to use SQL queries as a malicious input.

TALK ABOUT OTHER SAFETY FEATURES I COULD USE IN MY CODE – DON’T HAVE TO USE THEM BUT GOOD TO TALK ABOUT THEM AND THEIR POTENTIAL BENEFITS

For the most part, creating an entirely secure search platform isn’t the main focus of my project. I will of course add in basic functionalities to keep any data that I extract safe and out of the hands of any unauthorized persons but the overall security of the data that is stored will be down to the effectiveness of the server. Additionally, the program will most likely only be used on a small scale and so there isn’t the concern that the data will get into the wrong hands under any circumstances. For this reason, with the exception of PDO (AND MAYBE OTHER EXTENSIONS I FIND AND USE), the information that I talk about in this section is outside the scope of the project.

## 2.6 AI Methodologies

A neural network is one of the most common ways in which large amounts of data is processed in order to be categorized. The definition of a neural network is ‘a computer system [modelled](https://www.google.com/search?sca_esv=584679428&rlz=1C1ONGR_en-GBGB973GB973&sxsrf=AM9HkKmMqmwpzB82FTQpiyJQ0IlVkTAtYw:1700689586941&q=modelled&si=ALGXSlaYxyllm14_NEvUA9w95SVc_tVM6yCh0a6ETiP24ywJ1DtcNNYJsZHBAkufdSVv_Le8X8CYdblHB0Stb6mMoelM7lpVOvmRBMiu9io_bOWe-2tlsTs%3D&expnd=1) on the human brain and nervous system’ (*Oxford Languages and Google - English | Oxford Languages*, no date), meaning that there are a large number of nodes (neurons) that are interconnected with one another to pass on data. The data is processed in such a way that a value is returned at the end that can be used as a prediction or classification for the input values that were given (Abiodun *et al.*, 2019). The first mathematical model, called the MP model, was devised by McCulloch and Pitts 1943 but never taken seriously because of its limited capabilities. Rosenblatt proposed the single layer perceptron which was based on the MP model but added learning capabilities (much like the human brain) but this new model still couldn’t handle linear inseparable problems. Rumelhart et al. propositioned a multilayer feedforward network that is trained by the error backpropagation algorithm, and this formed the basis for all neural networks that we use today. (Li *et al.*, 2022)

A convolution neural network is a specialised form of neural network that operates most effectively on images. This has made it revolutionary for computer vision technologies such as face recognition, autonomous vehicles and intelligent medical treatments. However, this area of the topic is out of the scope of the project because there is no form of image recognition.

However, it is important to understand that neural networks come with potential drawbacks. For instance, a significant amount of pre-processing is required in order to make a data set useable by the model. This process involves data cleansing to fill in any missing or erroneous values, removing outliers, and data normalisation. The data must then be split into a training, validation and test set, and the way in which the data is split is very important because it could lead to skewed data in one or more of the sets causing one of the phases to fail. Any or all of these processes can be automated by some algorithm and this is especially common when vast dataset are being used because it would be too time consuming for a person, or group of people, to try and sort through all of the data manually. However, this places a lot of trust on the algorithm which is not always sensible because any such algorithm is never guaranteed to perform perfectly without causing errors, and so a human would have to check through all of the data after it has been processed anyway to check for any errors (Mishra *et al.*, 2020).

After the pre-processing phase, a suitable model has to be developed in a way that produces the best output with the majority of data that it is given. This process is very computationally expensive and time consuming because it requires a trial and error approach to be taken in order to find the best configuration of nodes, edges and weights (Livingstone, Manallack and Tetko, 1997).

Due to the complexity of a neural network, it can often be almost impossible to analyse the way in which the output is calculated. The lack of knowledge for the causal relationships between nodes in the network can lead to misleading interpretations and predictions, especially in large and complex networks. The number of edges is dependent on the architecture of the network, but also dependent on the number of nodes and hidden layers used for some computation in the network. Let N=the number of nodes in the input layer, M=number of nodes in the output layer and L=the number of hidden layers (each with H[i] nodes) then the total number of edges in a typical neural network is

E = N × H[1] ​+ (∑ (i = 1), (L − 1​) H[i] ​× H[i + 1]) ​+ H[L] ​× M.

As shown, as the number of nodes or layers increases, the number of edges increases at a faster rate. This is a problem with very large and complex neural networks because the computation time increase exponentially, meaning that a large number of resources are needed for a project involving a large neural network (Garg, Jegelka and Jaakkola, 2020).

Furthermore, it is common for a neural network to lack in robustness, meaning that the network may not adjust well to different inputs of data. For example, if the model is trained on a certain dataset then it will only be familiar with the data in the set and other data points of a similar nature, so if a new data point is given to the model that doesn’t fit within the normal ranges of the training data, then the network may incorrectly process or classify it. Similarly, with the user of image processing in a neural network, a small change to an image, like a change in the lighting, wouldn’t change the perception of a human, but the network would be unsuccessful at processing the image if it hadn’t been tested on an image with this level of lighting before (*What are the limits of deep learning?*, no date).

Another area of deep learning that is commonly found is Bio-Inspired AI. This field studies the behaviour of individuals and groups of animals found in different environments in the world to find new ways to go about solving problems. The Particle Swarm Optimisation method draws inspiration from the behaviour of a flock of birds trying to hunt a school of fish. As opposed to one bird trying to catch prey, a flock has the ability to search through a large search space and alert the individuals around them of any successes that they have during a certain time. This process can be replicated as a method used by an AI algorithm to find the optimal value in a search space. A ‘swarm’ is initialised and each particle is allocated a random position and random velocity in the search space. At each increment, the position is updated based on the velocity of the particle, and a utility function is applied to evaluate how good the current position is in relation to some target utility. Each particle then looks at the utilities of it’s neighbours and updates it’s velocity in relation to closer areas of the search space that seem to give a high utility. After a large number of iterations, the particles will find an optimal point in the search space, this is known when the particle no longer has a velocity. This idea of group behaviour has many more applications and shows how emergent properties in a group are demonstrated by each individual learning and helping other individuals around them.

## 2.7 AI Uses in Web Browsers

The use of AI libraries and APIs plays a key role in the implementation of any machine learning related program due to the ease and vast capabilities that they provide. With a search engine, automation can be introduced into the system in a number of different areas in order to improve the user experience, increase security and maximise efficiency. For example, the use of machine learning algorithms to scan the content of webpages for threats (already discussed).

As a way to add convenience to the searching process for the user, a natural language processing (NLP) model can be used to scan the content of a website to create a summary that can be shown to the user. This way, a user can know what a webpage is about without having to spend time reading through it and therefore can save time and effort. One of the main NLP models is called BeautifulSoup which creates a parse tree for parsed webpages, giving the developer the ability to extract data from the HTML document. This makes it very useful for web scraping, allowing the developer to use elements in an HTML document as pieces of data to be processed. Other NLP models include LXml and RegEx with RegEx being the fasted but least scalable due to having limited extraction rules (Thivaharan., Srivatsun. and Sarathambekai., 2020).

Applications of neural networks in web browsers are moderately common, with the vast amount of data stored on a person on the Internet, it can be helpful to create ways to process the data in ways that benefit both the user and content providers. A single user will mostly likely have countless accounts on network services on the Internet and so linking these accounts into one, central area can be convenient and efficient for a user. This process is called identity linking (Qiao *et al.*, 2020). Identity linking has a number of other useful benefits as well such as grouping users into group with something in common. This could be identifying that a user is a minor and so should be restricted from certain content, or identifying a malicious user that intends to do harm to another user, so it is in the interest of the browser provider to stop this to prevent a cyber-attack on one of their users. In the study by Qiao et al. a Siamese neural network was used compare high level feature representation of the behaviour of web users. MAYBE ADD MORE – LINKS VERY CLOSELY TO MY PROJECT

A study by Kolakowska et al. sought to determine whether it is possible to track human computer interactions like keystroke dynamics and mouse movements to recognise the gender of a web browser user. Other studies have been made on the use of these human computer interactions, for example, recognising a user from their mouse movements (Pusara and Brodley, 2004), or detecting cognitive and physical stress through typing behaviour (Vizer, 2009), but none have gone as far as to accurately predict a key personal characteristic like a person’s gender. The experiment involved recording the interactions with a mouse, keyboard, track pad and touch screen but only enough meaningful data was collected for mouse movements after the pre-processing phase. Results showed that the user of a rotation forest to process and categorise new data points yielded an 80.8% success rate and a Bayesian Network gave a 79.12% success rate.

OTHER AI TECHNOLOGIES FOUND AND USED LATER ON.

## 2.8 Development Process Models

## 2.8.1 Waterfall

The waterfall development lifecycle is one of the simplest models created. The different steps are definition, design, implementation and unit testing, integration and unit testing, and deployment. Each step in the cycle must reach completion before the next to start, and once they have been marked as finished, it is not possible to return to that stage. This makes the waterfall mode static which means that it can’t adjust to changing requirements. Its main uses are in very large projects because it can be useful to have a set list of requirements that the development team can work towards, and they can be confident that what they are making is exactly what has been previously described to them. However, other models can be a better fit for smaller teams. (Klopper, Gruner and Kourie, 2007), (Lock, no date)

## 2.8.2 Spiral

The spiral model develops software in a flexible and step-by-step manner, by repeating cycles, or "spirals," to gradually build and improve the software. Each cycle involves four key steps: setting goals and identifying risks, finding solutions and planning the next steps, building the software, and testing and evaluating the results. After each cycle, the new findings and learnings are used to update and improve the software. This process continues until the software meets all the requirements. The spiral model is like climbing a staircase, where each step brings the software closer to being finished, and any issues are addressed along the way. (Klopper, Gruner and Kourie, 2007), (Lock, no date)

## 2.8.3 V Model

This model uses similar ideas to the Waterfall except it highlights the relationship between each phase of development and its corresponding phase of testing. The left side of the "V" represents the development phases, and the right side represents the testing phases. The advantage of the V-Model is that it provides a structured approach to testing, ensuring that testing activities are planned and conducted at each stage of development. It also emphasizes early testing to catch and fix defects as early as possible in the development process. (Klopper, Gruner and Kourie, 2007), (Lock, no date)

## 2.8.4 Incremental

The system is divided into small manageable parts where each part works towards the overall functionality of the system. Development is performed incrementally, and at each increment new features are added, or existing parts are improved. This approach allows the gradual improvement of the system to be shown to shareholders and gives the opportunity for regular feedback that can be added into the next increment, which is especially good when the requirements aren’t clear or initially misunderstood. However, this comes with increased costs and will be harder to manage. Additionally, it is more time consuming because in depth documentation is needed for each increment. (Klopper, Gruner and Kourie, 2007), (Lock, no date)

## 2.8.5 Agile

Agile development aims to make software in a flexible and collaborative way. Instead of following a rigid plan, it focuses on working closely with people and being open to change. The main idea is to deliver small and functionally useful parts of the software in regular intervals which is usually every few weeks. This allows improvements to be made quickly based on the feedback from users. Teams include different types of experts that communicate frequently and use methods like Scrum or Kanban to organize their work. They have regular meetings to discuss progress and solve problems together. The goal is to create a product that fits the needs of users better and is more adaptable to changes. (Leau *et al.*, no date), (Lock, no date)

# 3 Abbreviations and Acronyms

AI - Artificial intelligence

NCSA – National Centre for Supercomputing Applications

# 4 Additional Transferable Skills

# 5 Project Journey???

Talk about the steps taken in completing my project and the report. Or will this just be spread amongst all the other sections?

# 6 Design

## 6.1 Overview

The program is split up into two main sections, namely the web crawler and the search engine. They run independent on one another in terms of when they are run because one deals with the traversing of the internet to build up a crawl list and the other deals with a user query to return a list of relevant websites.

The web crawler will run at different times to when there are users trialling or using the system to ensure that the database of crawled websites is always up to date with any new websites or is aware of a removal of a website from the Internet. It is important that the database is kept up to date because storing the URL for a website that doesn’t exist would cause an error when the user tries to access it. In alignment with the typical design of a web crawler, the program will be designed to follow links, traversing from webpage to webpage, making a record of the URL and other important information about each one. This information will be stored in the crawl database that is accessible to the search engine so that relevant webpages can be found based on a user’s query.

No interface is required for this part of the project because it’s main purpose is to create and populate the crawl database with website entries. However, it is helpful to produce outputs when am important section of computation has been completed in the terminal, this includes

The search engine is the part of the program that allows the user to interact with the crawl list and find information from websites based on the queries that they enter into the search bar. A simple user interface is important so that there are no unnecessary distractions for the user, and will consist of a title, search bar to input text, a submit button and a settings section. Upon submitting a search, the database will be queried for any matches and these will be returned and presented to the user in a clear and consistent manner. It is important that the user gets the most relevant information from their search and so the returned list will be sorted by their relevance score in relation to the query (this is based off the title and any keywords that are associated with the page) and displayed from first to last. Using pagination will further add to the minimal design of the page by only showing a small subset of the websites to the user at a time, ensuring that there is never an unnecessary amount of data on the screen at one time. It will be possible to navigate through the ‘pages’ of websites using controls at the bottom of the website.

A settings popup will also be added to allow the user to customize the program in a number of different ways. Firstly, the option to login or create a new account will provide the most functional changes because this will give the program access to any previous data stored about the user, hence allowing AI to create recommendations and tailor the experienced based on saved data. Without this data, the program will be at its default state, just a simple web browser. The user data will be stored in a number of databases on the server, one for basic user information such as name, password and age, and other databases will store the recent activity and interactions of the user. SQL queries and machine learning algorithms will bring this data together in meaningful ways so that additional functionality can be given to enhance the user experience. For example, altering the order in which relevant websites are displayed to the user based on the previous websites that they’ve accessed, or offering similar search query examples on related topics.

In compliance with ‘the right to be forgotten' in the GDPR, the settings menu will provide an option for the user to delete their account if they want to. If they are logged in, then the program will delete any data associated with the user from all databases. If no user is logged in then this feature will be disabled because account deletion is not possible with no username to refer to.

## 6.2 Privacy Settings

It is important for a webpage to give the user full control over their data and how it is stored by the company, therefore a necessity for the design is inclusion of a privacy settings menu. By giving a user this level of control, they will feel more comfortable using the service and find the program more trustworthy. This will be located in the settings menu and once the user clicks it, a popup will appear in the centre of the screen that explains the different settings that are available to them, and then benefits potential drawbacks of using or not using them. Firstly, there will be the option to disable specific cookies. Obviously it is not possible for some permutations of allowed and disallowed cookies to be used because they work together to build a profile of the user and collect other useful information. Cookies that a user will be able to disallow include all non-essential cookies, the tracking of their search history, their username, MORE. A toggle switch system will be implemented to allow for these options to be displayed in a clear and obvious way.

## 6.3 Search History

This will be a separate webpage that the user is taken to when they click the ‘search history’ button that is in the settings menu. It is called using the GET operator which allows the username to be sent in the URL of the website and used in an SQL query to get all search history entries for the particular user. Once this data has been retrieved, it will be grouped into days and sorted by it’s timestamp so that the most recent history is shown to the user first. MAYBE USE PAGINATION SO THAT NOT ALL DATA NEEDS TO BE LOADED AND DISPLAYED AT ONCE

Additionally, a delete button is included with all of the website entries, creating the option to quickly delete any entries that the user wants. If one of these is pressed then a query will be sent to the database, removing the whole row, and the webpage will be reloaded to show the deletion instantly.

## 6.4 Database Designs

### 6.4.1 Users

This database stores personal, identifiable information about registered users of the system and contains the columns: first name, last name, username, password, date of birth. In order to maintain the integrity of the database and ensure it is in third normal form, the primary key is the username column. All other columns have the possibility of being the same but a username for the system should be a unique identifier for a specific user, meaning that if personal data is every changed, then the entry is located using the username.

Database Operations:

* Register – Create a new row in the database that contains the first and last name, username, password and date of birth given by the user in the registration form. Uses a post operation to get all of the variables from ‘SearchEngineFrontEnd’ and performs an ‘INSERT INTO’ query on the users database to add the data. Returns a success or error message after the query has been carried out or attempted on users.
* Sign In – Performs a check on the users database using the username and password given by the user by querying ‘SELECT username, password FROM users WHERE username = :username’. If a row is returned, showing that this particular username exists, then the given password is compared to the password that is stored in users. If there is a match then a success message is sent back, allowing the user to be signed in in the website. If there is no match then the user is told that the username or password is incorrect and they cant be logged in until they amend the credentials.
* DeleteAccount – Accessed through the settings menu when a user is signed in. Retrieves the username from the webpage using a POST operation and queries the database with ‘DELETE FROM users WHERE username = :username’. This query will always be successful because a user has to be signed in to access this functionality and the username saved as a cookie in the website is guaranteed to be correct, however there is still some code for validation just in case any errors do occur in the deletion process.
* GetColumnEntry – Used to get personal information about the current user that is displayed when the personal information section is accessed in the settings menu. Data that is needed depends on the JavaScript function that calls it and so ‘columnName’ is given as a parameter and this determines which field of the users database is returned to the program. The value of columnName is username, fname, lname or password, all of which the user has the ability to change in the personal settings sections. The other parameter that is sent is the username of the current user, together with the column name, a SELECT command is sent to users that asks for the value from the column that relates to the username. This is returned and used to fill in a section of information about the user.
* ChangeColumnEntry – Uses a similar approach to GetColumnEntry because the column where the data belongs is specified in the parameters rather than being constant. In addition to columnName and username, a POST operation is used to get ‘newValue’ that gives will be used to replace the piece of data that the user has requested to change. ‘UPDATE users SET $columnName = :newValue WHERE username = :username” tells the database to change the value at column ‘columnName’ and row ‘username’ to ‘newValue’.

### 6.4.2 Websites

This database stores all of the websites that have been traversed by the web crawler. The columns consist of a title, keyword, URL, summary, relevance score and PageRank with the URL and keyword columns being combined to make the primary key. The reason for this is because a website is associated with a list of keywords but only one keyword can be stored per field meaning that some entries will only differ by a keyword but a keyword alone does not identify a website. Upon selecting data, entries need to be brought together to create one instance of a website that consists of a list of keywords. An instance of a website is formed using the object ‘Website’ consisting of attributes made from each column of the database table.

Database Operations (Python):

* SaveToCrawlList – Called when the crawler has reached the maximum depth for the current crawl cycle. At this point in computation, the program has created a list of Website objects that contain all of the data that needs to go in the website database. A SELECT query is sent to the database to check if a crawled website has already been saved in a previous crawl, if it has then there is no need to store the website again because it would lead to a duplicate entry in the database. If the website doesn’t already exist in the database then an INSERT INTO query is performed so that the URL, title, keywords and summary that have been retrieved or created by the program can be stored. An important feature of this operation is that all of the keywords in a website’s keyword list have to go inside all of their own entry in the database so that database integrity is maintained. This means that for any given website, the number of entries added to the database is equal to the length of the keywords list, and each of the entries only differs by it’s keyword.

Database Operations (PHP):

* OpenCrawlList – Called after when RetrieveWebsiteResults.php is first ran because the webpage requires the list of websites to display to the user. Uses a SELECT command to retrieve all websites stored in the websites database (crawl list). Then goes through each of the rows of data and extracts the data so that they can be used to create an instance of the Website class and added to $websiteList that is returned to RetrieveWebsiteResults.php so that the websites can be processed and a decision can be made on whether or not they are a close enough match to the user’s search query.

### 6.4.3 Search History

This database stores the URL, time visited and the username of who made the search. It has a composite primary key consisting of the URL and the time visited and the username is a foreign key linking the search history database to the user database which allows a user to view their specific search history and delete any entries if they would like to. Additionally, the URL acts as a foreign key that links this database to the website database, this makes it possible to extract the title of each website that is shown to the user when they are viewing their search history.

Database Operations (PHP):

* LogWebsiteVisits – Called using an onclick listener to check when a user visits a new website (clicks a link) while using the search engine. Extracts the clicked URL, clicked title and then username of the current users and queries the websiteHistory database using an INSERT INTO command where the data corresponds to the information about the click action, including the timestamp of the click. The username is required so that the visit can be linked to a particular user and be shown when they want to view their search history.
* GetWebsiteVisits – Uses a GET operation to get the username of the active user from the home page of the program and uses a SELECT command to get all website entries that match the extracted username. Once this list has been retrieved, each row is uses to create a list called ‘$entry’, consisting of the URL, title, date username and entry ID, and added to another list called ‘$data’. $data is then encoded into json format using ‘json\_encode’ and is returned back to the SearchHistory webpage where it can be processed and used to create the display for the user.
* DeleteSearchHistoryEntry – Called whenever the user clicks on the delete icon of one of the search history entries. Uses a POST operation to get the ID of the website which corresponds to the website’s ID that is used as the primary key in the database. After this, a DELETE command is sent to searchHistory.db in order to delete the entry of the website. A success or error message is then sent back to the client to specify if the query was a success or if there was an error.

## 6.5 Classes

### 6.5.1 Website

This is a critical class for the program because it allows all website data to be collected and stored in a way that is efficient and enables simple, consistent access. It is used in both the Python and PHP parts. More specifically, when the web crawler visits a website, all of the necessary data (URL, title, keywords and a summary) is extracted and used to create a instance of Website that can then be stored into the webpages database. In a similar sense to this, the search engine uses the Website class when it reads websites from the crawl list so that each row can be stored as an equivalent website, allowing the different functions to be applied to the data when needed.

The Website class contains the attributes URL, title, keywords, summary, relevance score and pageRank. Each of these have their own getter and setter methods. Additionally, for the PHP version of the class, a function is needed to calculate the relevancy score of the webpage. This is only needed at this stage of computation because no search query exists untul the user enters and submits it to the search engine, hence there is no possible way for the web crawler to calculate this value. calculateRelevance takes as input the search query from the user and firstly checks if the given website has already been allocated a relevance score. If yes then simply return the value, if not then more computation is needed. The score is set to zero initially, and then different scores for the title, keywords, URL and summary are calculated. For the title, FILL IN WHEN THE METHOD FOR SCORING EACH SECTION HAS BEEN FINALISED – ATM ITS TOO SIMPLE.

# 7 Implementation

## 7.1 Web Crawler

This section is implemented using Python. The reason for this is because it is one of the programming languages that I am the most confident in using, especially when it comes to tackling large projects with lots of different parts involved. Additionally, Python offers a large number of AI libraries that were very useful in creating important functionalities.

The program starts out by using the “requests” library to GET SPECIFIC USAGE OF REQUESTS AND HOW IT WORKS. Alongside this, a user agent is created that MORE ON USER AGENT

TALK MORE ABOUT IMPLMENTING THE WEB CRALWER – BUT NEED TO HAVE A MUCH MORE FINSIHED VERSION TO DO THIS

## 7.2 Search Engine

Because the purpose of this project is to provide an intuitive web browser for a user, the only option was to create the search engine using web based programming languages. More specifically, HTML, JavaScript and PHP. Doing this allowed a graphically pleasing and easily understandable user interface to be created in relatively simple ways. In the initial stages of planning, I contemplated the idea of creating the front end design in Python using the Tkinter library to make a pop up window that would give the user an interface to interact with and use the functionalities of the program. The reason for this was to keep the whole project Python-based which would have allowed more simplicity and consistency when it came to processing and using the data for various sections. However, it quickly became apparent that an approach like this would not be possible and instead I would have to make a web-based user interface that accessed the crawl list to give results for a user’s query. This is because it would have been overly complicated to try to handle the links for websites shown to the user and overall, the program would have been very slow due to the fact that it would have had to read lots of data from websites and then output them to the user. Hence, the obvious choice is creating a typical web browser interface that would allow the user to interact with the data in an easy manner.

The user interface follows a very similar structure to many the typical design of modern web browsers described in the literature review. The majority of the design is created from HTML code because the features are constant no matter the current state of the program. More specifically, the search bar is always positioned in the middle of the screen and the settings menu is located in the bottom right, these features never change.

### 7.2.1 Register

When the user creates a new account using the ‘register’ option, their username, password, first name, last name and date of birth are sent to a PHP script called ‘Register.php’ using a POST operation. From here, an SQL query is sent to the ‘users’ database to check if a user already exists with the given username, if so then an error message is sent back to the user telling them that their username needs to be unique and so they will have to choose another one. If the username is unique and all of the other data has been filled out in the correct format then another SQL query is sent to the database using the ‘INSERT INTO’ command so that a new entry is created and stored in the database. A confirmation message is sent back to the webpage, the user is logged in and the username cookie is rewritten to match the username of the current user.

### 7.2.2 Sign In

For a user to sign into an existing account, a similar method is used as with registering a new account in the system. Once the user fills in the username and password fields in the webpage, and they have been deemed to not be empty, a POST operation is sent to ‘SignIn.php’ that reads and stores the login credentials as variables that are used in an SQL query sent to the users database where they are compared to other entries. If there is a match then a success message is sent back to the user and they are logged in to the system. If not then an error message is sent back telling the user that the username or password that that they entered are incorrect.

### 7.2.3 Log Out

A div called ‘auth-status’ is placed in the settings div, below the settings button, that is initially empty but when the window is loaded, the login status is checked to see if any username is associated with the username cookie. If a user is logged in then ‘auth-status’ is filled with a welcome message and a button that logs the user out if pressed. In this case, the function ‘logout’ is called that sets the username cookie to empty and calls ‘checkCookie’ to refresh the authentication status back to not specified. This means that all parts of the program will know that no user is logged in and so certain features wont be available.

## 7.3 AI Data Extraction

### 7.3.1 Cookies

To enable the efficient collection and processing of user data, a number of different cookies are utilised during the time that a person interacts with the user. This process only happens with the consent of the user, which is explicitly requested the first time a device accesses the website.

Once the permission of the user has been granted, the first cookie that is saved is called “visitedBefore” which marks if the device has used the website before. The purpose of this feature is to let the browser know if the cookie request needs to be shown to the user before they use the services provided, if it is set to true then there is no need to request permission again. visitedBefore is stored on the clients browser for a year before it expires and at this point the permission to continue the use of cookies will be shown again. Addititionally, the suer has the option to reset this cookie back to false so that they can respecify all of the cookie permission that they have previously allowed.

In order to represent the user’s choice on cookie permissions, a cookie called “cookieConsent” is used. This cookie lets the rest of the program know whether or not a cookie can be saved to track some form of activity from the user. If this is set to false then the functionality of the program is significantly refuced because no personal data about the user and their search hostiry or style can be tracked, hence meaning that their data cant be processed by the AI elements. However, it is necessary to include this so that a user’s privacy can be respected, LAW FOR THIS. Together with visitedBefore, these two cookies make up the essential cookies of the website, which means that they are needed so that a basic amount of information about the user is known and useable by the browser.

A cookie called “username” is one of the most frequently occurring cookies in the program because it refers to the current user that is interacting with the system. It is set or changed when a new user signs in from their browser and set to null if this user signs out or deletes their account. It has a wide range of uses throughout all parts of the browser. For example, whenever a link is clicked, taking the user to a new webpage, this has to be stored in the search history database so that the information can be used or viewed later on, and this data is linked to the user so that it is personal. Another way in which username is used is to MORE USES OF USERNAME

## 7.4 AI Data Processing

## 7.5 AI Data Uses

# 8 Results

# 9 Interpretation of Results

Maybe can be combined with Results section??

# 10 Figures and Tables

# 11 Conclusion

# 12 Future Work

# 13 References

Abiodun, O.I. *et al.* (2019) ‘Comprehensive Review of Artificial Neural Network Applications to Pattern Recognition’, *IEEE Access*, 7, pp. 158820–158846. Available at: https://doi.org/10.1109/ACCESS.2019.2945545.

Ahn, J., Kim, K. and Proctor, R.W. (2018) ‘Comparison of Mobile Web Browsers for Smartphones’, *Journal of Computer Information Systems*, 58(1), pp. 10–18. Available at: https://doi.org/10.1080/08874417.2016.1180652.

Alharbi, J.A. *et al.* (2023) ‘An Empirical Analysis of E-Governments’ Cookie Interfaces in 50 Countries’, *Sustainability*, 15(2), p. 1231. Available at: https://doi.org/10.3390/su15021231.

*Browser History: Epic power struggles that brought us modern browsers* (no date) *Mozilla*. Available at: https://www.mozilla.org/en-US/firefox/browsers/browser-history/ (Accessed: 20 November 2023).

*Cookies, the GDPR, and the ePrivacy Directive* (2019) *GDPR.eu*. Available at: https://gdpr.eu/cookies/ (Accessed: 21 November 2023).

Das, A. (2022) *Does Disabling JavaScript Protect You From Hackers?*, *MUO*. Available at: https://www.makeuseof.com/should-you-disable-javascript-to-protect-from-hackers/ (Accessed: 21 November 2023).

Davitt, K. *et al.* (2024) ‘CoStricTor: Collaborative HTTP Strict Transport Security in Tor Browser’, *Proceedings on Privacy Enhancing Technologies* [Preprint]. Available at: https://petsymposium.org/popets/2024/popets-2024-0020.php (Accessed: 21 November 2023).

*Evolution of Dark Web Threat Analysis and Detection: A Systematic Approach | IEEE Journals & Magazine | IEEE Xplore* (no date). Available at: https://ieeexplore.ieee.org/abstract/document/9197590 (Accessed: 20 November 2023).

Garg, V., Jegelka, S. and Jaakkola, T. (2020) ‘Generalization and Representational Limits of Graph Neural Networks’, in *Proceedings of the 37th International Conference on Machine Learning*. *International Conference on Machine Learning*, PMLR, pp. 3419–3430. Available at: https://proceedings.mlr.press/v119/garg20c.html (Accessed: 23 November 2023).

Gregor, P. and Dickinson, A. (2007) ‘Cognitive difficulties and access to information systems: an interaction design perspective’, *Universal Access in the Information Society*, 5(4), pp. 393–400. Available at: https://doi.org/10.1007/s10209-006-0064-6.

*How do we comply with the cookie rules?* (2023). ICO. Available at: https://ico.org.uk/for-organisations/direct-marketing-and-privacy-and-electronic-communications/guide-to-pecr/guidance-on-the-use-of-cookies-and-similar-technologies/how-do-we-comply-with-the-cookie-rules/ (Accessed: 21 November 2023).

Hr, M.G. *et al.* (2020) ‘Development of anti-phishing browser based on random forest and rule of extraction framework’, *Cybersecurity*, 3(1), pp. 1–14. Available at: https://doi.org/10.1186/s42400-020-00059-1.

Kumar, S. and Sharma, R.R. (2015) ‘Empirical Analysis of Unethical Practice of Cookies in E-Marketing’, *Abhigyan*, 33(3), pp. 42–56.

Kyriakoullis, L. and Zaphiris, P. (2016) ‘Culture and HCI: a review of recent cultural studies in HCI and social networks’, *Universal Access in the Information Society*, 15(4), pp. 629–642. Available at: https://doi.org/10.1007/s10209-015-0445-9.

Li, Z. *et al.* (2022) ‘A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects’, *IEEE Transactions on Neural Networks and Learning Systems*, 33(12), pp. 6999–7019. Available at: https://doi.org/10.1109/TNNLS.2021.3084827.

Liu, C.-H., Chen, W.-K. and Sun, C.-C. (2020) ‘GUIDE: an interactive and incremental approach for crawling Web applications’, *The Journal of Supercomputing*, 76(3), pp. 1562–1584. Available at: https://doi.org/10.1007/s11227-018-2335-4.

Livingstone, D.J., Manallack, D.T. and Tetko, I.V. (1997) ‘Data modelling with neural networks: Advantages and limitations’, *Journal of Computer-Aided Molecular Design*, 11(2), pp. 135–142. Available at: https://doi.org/10.1023/A:1008074223811.

Marcus, A. and Gould, E.W. (2000) ‘Crosscurrents: cultural dimensions and global Web user-interface design’, *Interactions*, 7(4), pp. 32–46. Available at: https://doi.org/10.1145/345190.345238.

Mazher, N., Ashraf, I. and Altaf, A. (2013) ‘Which web browser work best for detecting phishing’, in *2013 5th International Conference on Information and Communication Technologies*. *2013 5th International Conference on Information and Communication Technologies*, pp. 1–5. Available at: https://doi.org/10.1109/ICICT.2013.6732784.

Mishra, P. *et al.* (2020) ‘New data preprocessing trends based on ensemble of multiple preprocessing techniques’, *TrAC Trends in Analytical Chemistry*, 132, p. 116045. Available at: https://doi.org/10.1016/j.trac.2020.116045.

*Oxford Languages and Google - English | Oxford Languages* (no date). Available at: https://languages.oup.com/google-dictionary-en/ (Accessed: 22 November 2023).

Pusara, M. and Brodley, C.E. (2004) ‘User re-authentication via mouse movements’, in *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*. New York, NY, USA: Association for Computing Machinery (VizSEC/DMSEC ’04), pp. 1–8. Available at: https://doi.org/10.1145/1029208.1029210.

Qiao, Y. *et al.* (2020) ‘Siamese Neural Networks for User Identity Linkage Through Web Browsing’, *IEEE Transactions on Neural Networks and Learning Systems*, 31(8), pp. 2741–2751. Available at: https://doi.org/10.1109/TNNLS.2019.2929575.

*Recital 30 - Online identifiers for profiling and identification* (2018) *GDPR.eu*. Available at: https://gdpr.eu/recital-30-online-identifiers-for-profiling-and-identification/ (Accessed: 21 November 2023).

Sa-nga-ngam, P. and Kiattisin, S. (2020) ‘Increasing Web Accessibility Through a Personalized Web Browser for Older Adults’, *Wireless Personal Communications*, 115(4), pp. 3235–3259. Available at: https://doi.org/10.1007/s11277-020-07220-6.

Shah, R.C. and Kesan, J.P. (2009) ‘Recipes for cookies: how institutions shape communication technologies’, *New Media & Society*, 11(3), pp. 315–336. Available at: https://doi.org/10.1177/1461444808101614.

Shakir M. Abass, K.J. (2019) ‘Development History Of The World Wide Web’. Int. J. Sci. Technol. Res, 8(9), pp.75-79. Available at: https://www.researchgate.net/profile/Karwan-Jacksi/publication/336073851\_Development\_History\_Of\_The\_World\_Wide\_Web/links/5d8d1f8f92851c33e94064cb/Development-History-Of-The-World-Wide-Web.pdf.

Su, Q.-Y. and Adams, C. (2005) ‘Will B2C e-commerce developed in one cultural environment be suitable for another culture: a cross-cultural study between amazon.co.uk (UK) and dangdang.com (China)’, in *Proceedings of the 7th international conference on Electronic commerce*. New York, NY, USA: Association for Computing Machinery (ICEC ’05), pp. 236–243. Available at: https://doi.org/10.1145/1089551.1089598.

Thivaharan., S., Srivatsun., G. and Sarathambekai., S. (2020) ‘A Survey on Python Libraries Used for Social Media Content Scraping’, in *2020 International Conference on Smart Electronics and Communication (ICOSEC)*. *2020 International Conference on Smart Electronics and Communication (ICOSEC)*, pp. 361–366. Available at: https://doi.org/10.1109/ICOSEC49089.2020.9215357.

Vidas, T. *et al.* (2013) ‘QRishing: The Susceptibility of Smartphone Users to QR Code Phishing Attacks’, in *Financial Cryptography and Data Security*. *International Conference on Financial Cryptography and Data Security*, Springer, Berlin, Heidelberg, pp. 52–69. Available at: https://doi.org/10.1007/978-3-642-41320-9\_4.

Virvilis, N. *et al.* (2015) ‘Security Busters: Web browser security vs. rogue sites’, *Computers & Security*, 52, pp. 90–105. Available at: https://doi.org/10.1016/j.cose.2015.04.009.

Vizer, L.M. (2009) ‘Detecting cognitive and physical stress through typing behavior’, in *CHI ’09 Extended Abstracts on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery (CHI EA ’09), pp. 3113–3116. Available at: https://doi.org/10.1145/1520340.1520440.

*What are the limits of deep learning?* (no date). Available at: https://doi.org/10.1073/pnas.1821594116.

*What Is Web 3.0 (Web3 definition)?* (no date) *What Is Web 3.0 (Web3 definition)?* Available at: https://www.avast.com/c-web-3-0 (Accessed: 20 November 2023).

Yu, L., Li, Y. and Zeng, Q. (2021) ‘Design of topic Web crawler based on improved PageRank algorithm’, *Journal of Physics: Conference Series*, 1754(1), p. 012210. Available at: https://doi.org/10.1088/1742-6596/1754/1/012210.