Using Client-Side Substitution Ciphers to Enhance Password Strength

Final Report

Student: Kevin Davitt
Supervisor: James Egan
C00215181@itcarlow.ie

Abstract

The purpose of this project is to create a Browser Extension that a user can use to transform text in password input fields according to substitution ciphers that are unique for each service a user may use. This enables users to use previously insecure password practices, securely. This enables convenience without compromising security. The application should be made of two components, a backend web server that stores all the required data such as the substitution ciphers, herein referred to as "maps", and provide general signup and authorisation functionality. The other component is the Browser Extension itself, which should provide the user with an interface to authenticate with the Web application and select the map to use with the corresponding service. If the user has not set up a map for a web service, there should be the option to generate a map and begin applying it to the desired service.

Contents

Introduction

This report will document the process carried out while designing the Password Mapper application, herein referred to by its marketing name "Mappa". Mappa is a password security service which enables users to use previously insecure password policies to improve authentication convenience without compromising security, for example, users can safely use shorter passwords and dictionary words securely. They can also reuse passwords across multiple sites, without compromising security. This is achieved using substitution ciphers linked to services. Throughout this project, a huge emphasis was placed on bringing a product to market with security in mind, in terms of the security of the application itself.

While there are like-minded applications available as discussed in the Research Manual, they all use the standard password manager approach, which is storing passwords for user services in a password vault and granting the user access to this vault when the user authenticates with the password manager software. The difference with Mappa is that, while users must still authenticate with the Mappa service, Mappa does not store users passwords and instead stores a unique cipher for each service a user wants to use Mappa with. The user must then also remember a short seed word to combine with this cipher to produce their password for a service. These seed words can be the name for every service or different, at the user's discretion. Even if seed words are the same, the cipher is different and so will produce a different password. This provides the convenience of password reuse for the user without the security concerns.

The first section of this document outline the description of the final submission of the project. It contains an overview of technologies used and a detailed description of each screen and its functionality. This section will also discuss some difficulties faced in development, both solved and unsolved.

The next section will show how the final project compares to the original functional specification and design specification for Mappa. It will explain details that have changed during the development process and what functionality has been added or excluded. This section will also discuss changes to the layouts of both components of the application, compared to the original design.

Additionally, this document will contain learning outcomes for this project, from both a technical and personal standpoint. This section will also analyse the technologies used more thoroughly. Technical and personal learning outcomes including time management and milestones will also be explored in this section.

The final section of this document presents an overall review of the final project, including both the successes and failures involved in the development of Mappa and what would be changed if there was an opportunity to start from the beginning. There will also be a brief outline of what the next steps would be going forward.

Project Description

Mappa was built in two components using Visual Studio Code as it's development environment. The two components are the Web server, which was run on Windows using XAMPP to host a local webserver. The second was a Browser Extension which was developed with Google Chrome in mind but also compatible with Firefox. Gitlab was integrated with Visual Studio Code to provide source control throughout the duration of the project. Google Chrome was used for all development, debugging and most testing. Google provide support and documentation for developing extensions for Chrome on their website.

The backend of the Webserver was developed using PHP, and the final product was hosted using an Amazon Web Services (AWS) EC2 instance. There were a number of components that had to be configured to get this working successfully, including the installation of a LAMP server on the EC2 instance, hardening of the LAMP software, configuration of load balancers, configuration of DNS records, and configuring TLS Cryptographic suites along with obtaining the required digital certificate from a Certificate Authority. The database on this server was a Maria Database, this was configured and use to store the details of user accounts, maps for users, logging, session information and security requirements such as device tracking and cross site request forgery tokens. Details of these tables can be found in the Design Manual.

The Website front end was designed using HTML5, CSS and JavaScript. Popular CSS Framework Bootstrap and icon library Font Awesome were also used to aid with the styling of HTML pages. The Browser Extension also utilised these web technologies. The Browser extension also heavily used AJAX requests to reach out to the server to retrieve user maps and other information, such as how many login attempts the user had remaining before being logged out. This information was sent to the extension from the Web server in JSON format. Similarly, HTTPS POST requests were used to enable users to log in and create new maps from within the extension popup window. A JSON manifest file is also required in order to communicate the browser permissions required by the extension to the browser.

To summarise, configuration of the Web server involved a LAMP server running in the cloud hosted by AWS and hardening of this server. Installation of the Browser Extension involves pointing a browser, in most cases Google Chrome at the folder containing all extension's files. The extension could also be simply downloaded from the Chrome Extension Store.

Application Screenshots
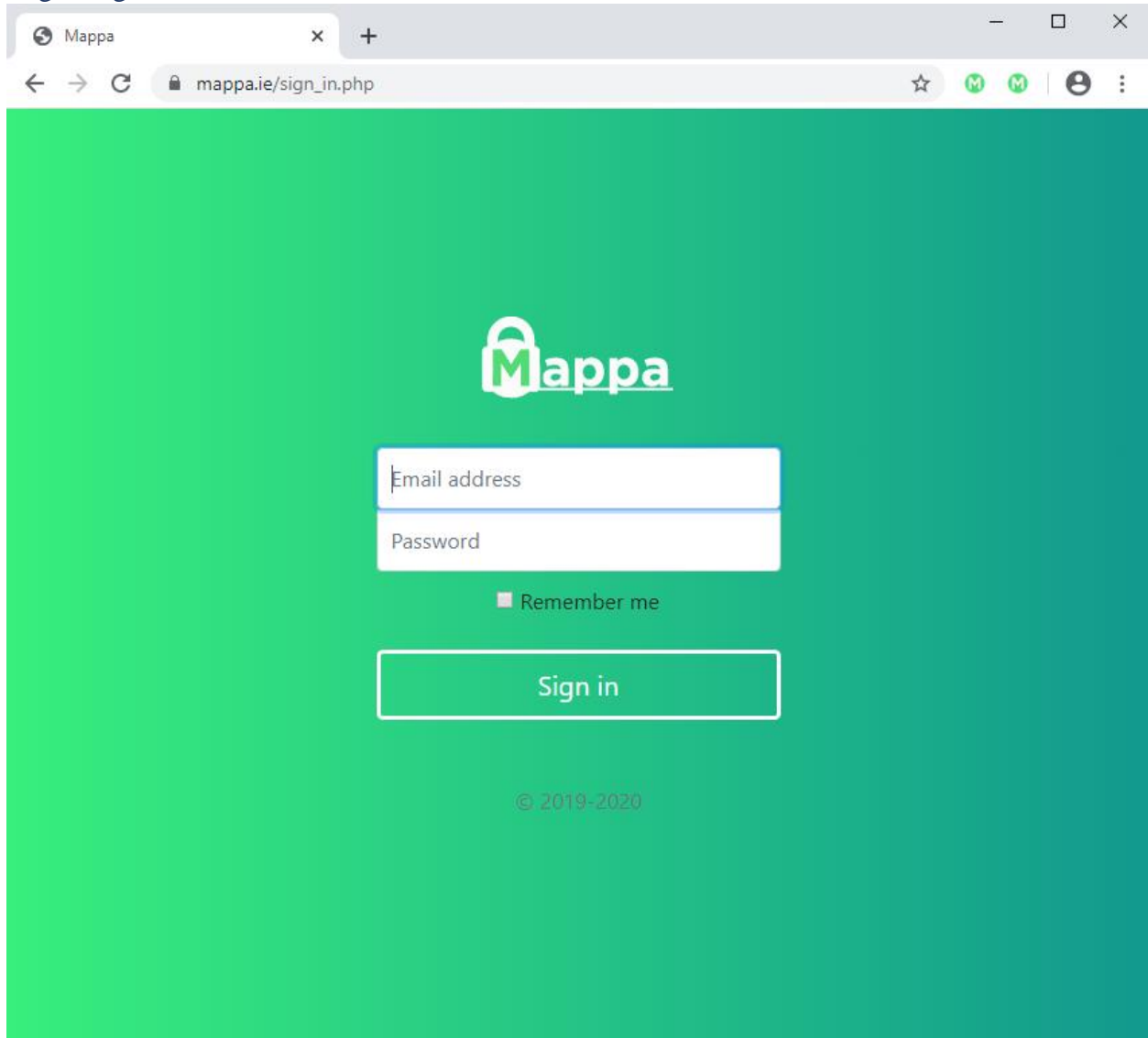
Mappa Web Server

Home Page
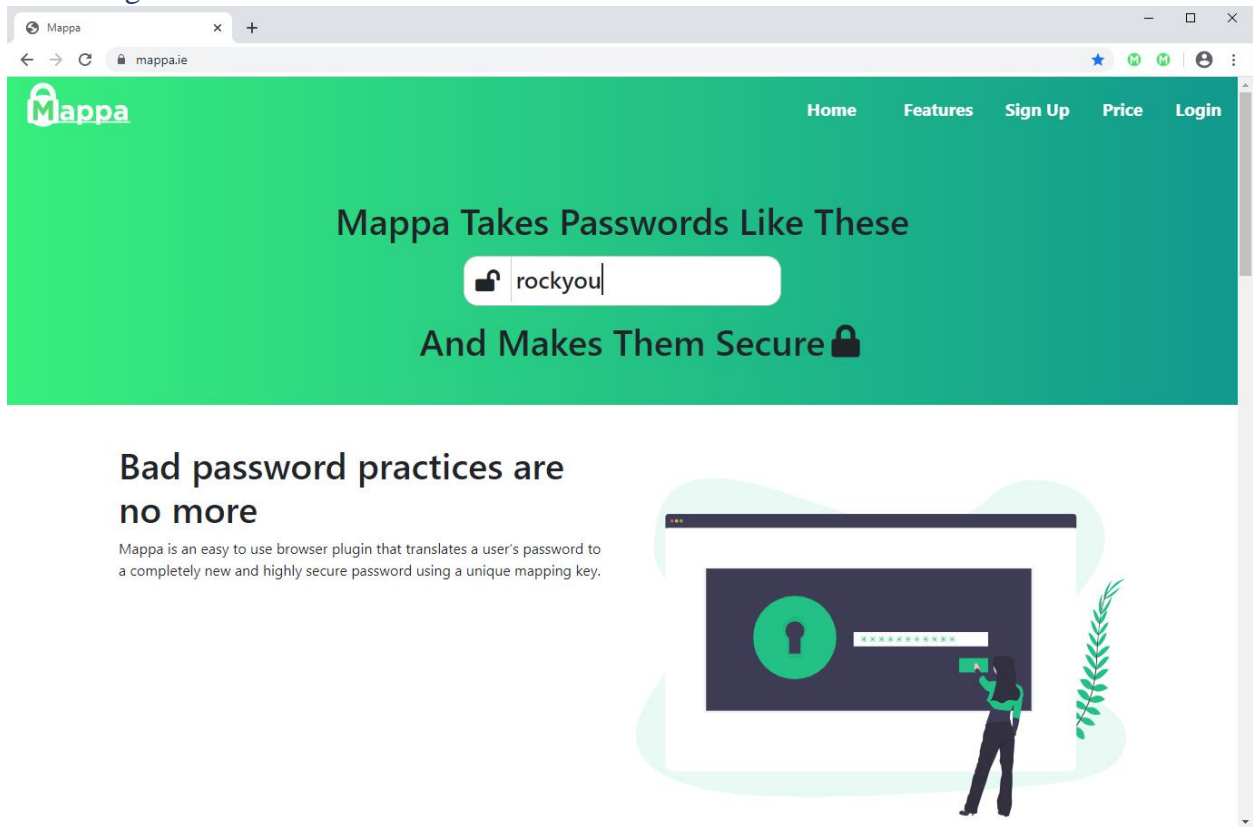
This home page can be visited at https://mappa.ie/home.php, it is much longer than the screenshot shown here and provides the pitch for this project as a viable business product. This page is used to navigate to the Sign Up, Log In and Account Options pages.

Login Page

This login page is used to authenticate with Mappa on the website itself. The Log In option on the site is only required when wishing to navigate to the account options page, where a user can perform account security related functions such as update their password. This page is resistant to brute forcing through the use of device tracking (user agent and IP combination), and protected from Cross Site request forgery (CSRF) through the use of an anti-CSRF token as a hidden field in the form. This CSRF token is attached to the device identifier.

Sign Up Page

The Sign Up page is used by a prospective user to create an account with Mappa. This page is also protected from CSRF like the above. This page also contains a brief explanation has to how to get using the application.

Account Options Page
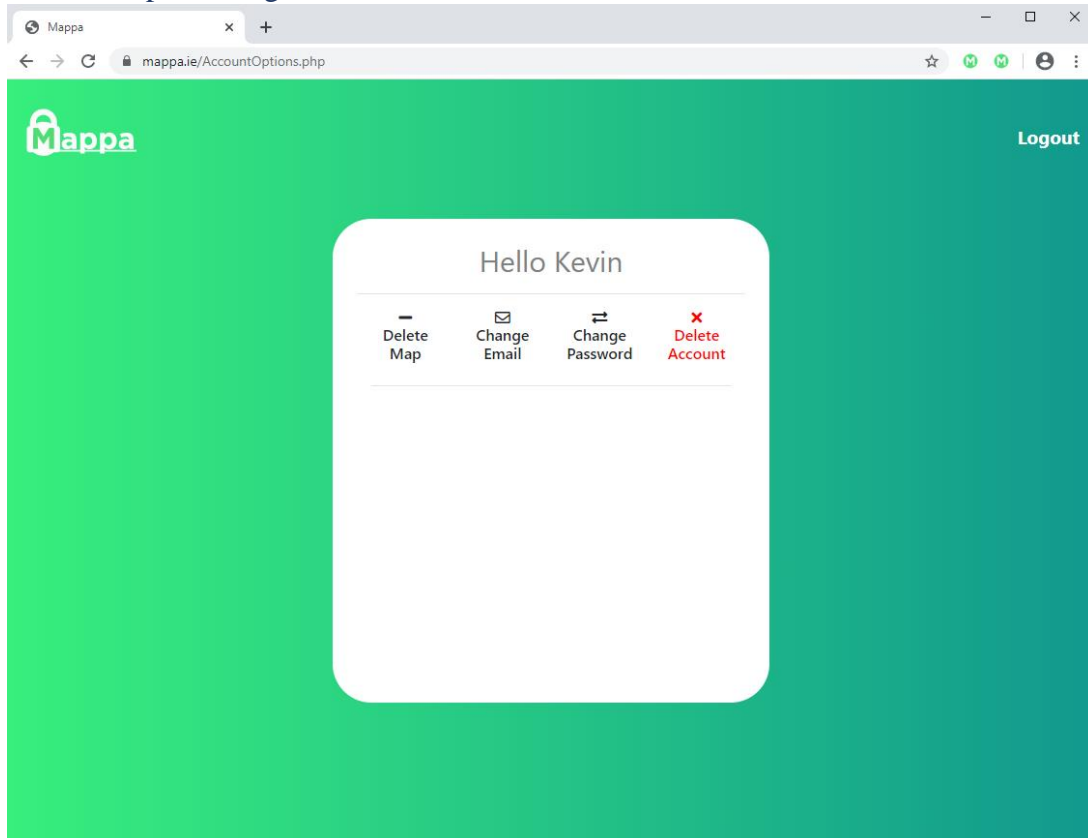
This is an authenticated page and enables users to carry out the functions listed. Delete Map allows a user to delete a map associated with their account, this is not a very common action but may be necessary under specific circumstances. Change Email and Change Password are self-explanatory. Delete account allows a user to totally delete their account and all their related data.

When a button is clicked the corresponding form appears in the whitespace below. Each function here is a security related function and as such requires the user to reauthenticate before the Server will execute the function. An example is show below for each outcome of the delete map option.
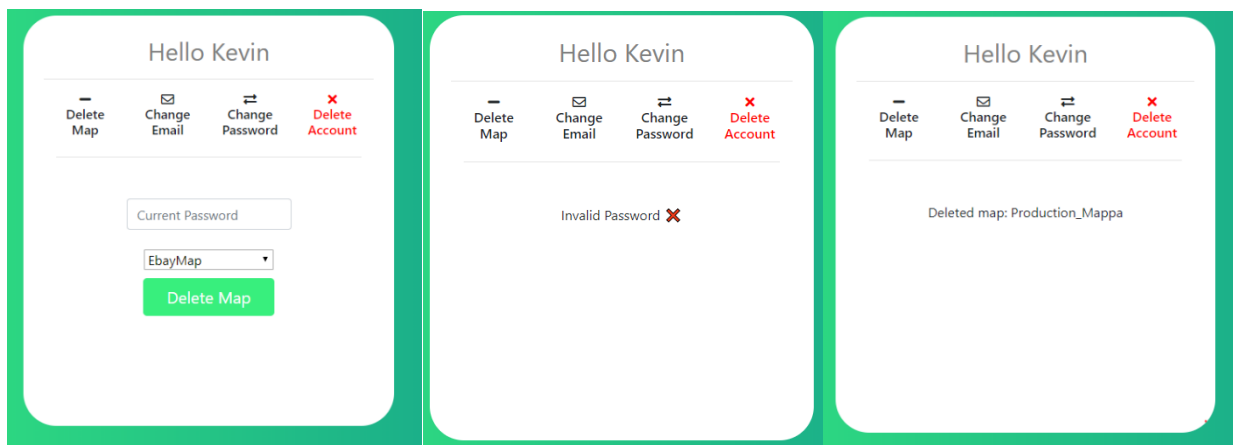
Other Functions

Good session management practices are taken throughout the Web Application. For example there is a log out button on each authenticated page. When the user presses this button they are securely logged out and their session destroyed. The headers on the Apache server also specify that the browser should not cache pages and so pressing the back button will not result in information disclosure by showing previous authenticated pages.

Web Extension

Login Page
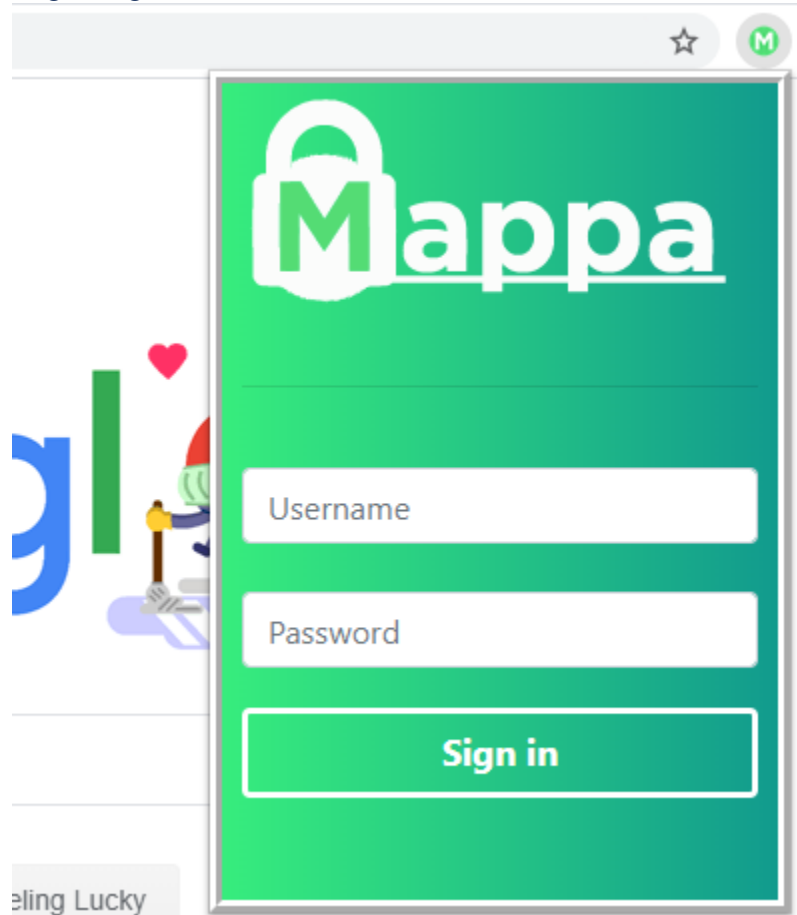
On clicking the Browser Extension icon at the top right of the page, the following Popup appears. A portion of the browser window has been left in the screenshot to lend context to the size and overlaid appearance of the window. This page function is identical to the Log In page on the Web site but is more convenience for the user to log in on. This page also implements CSRF protection.

Map Password Page
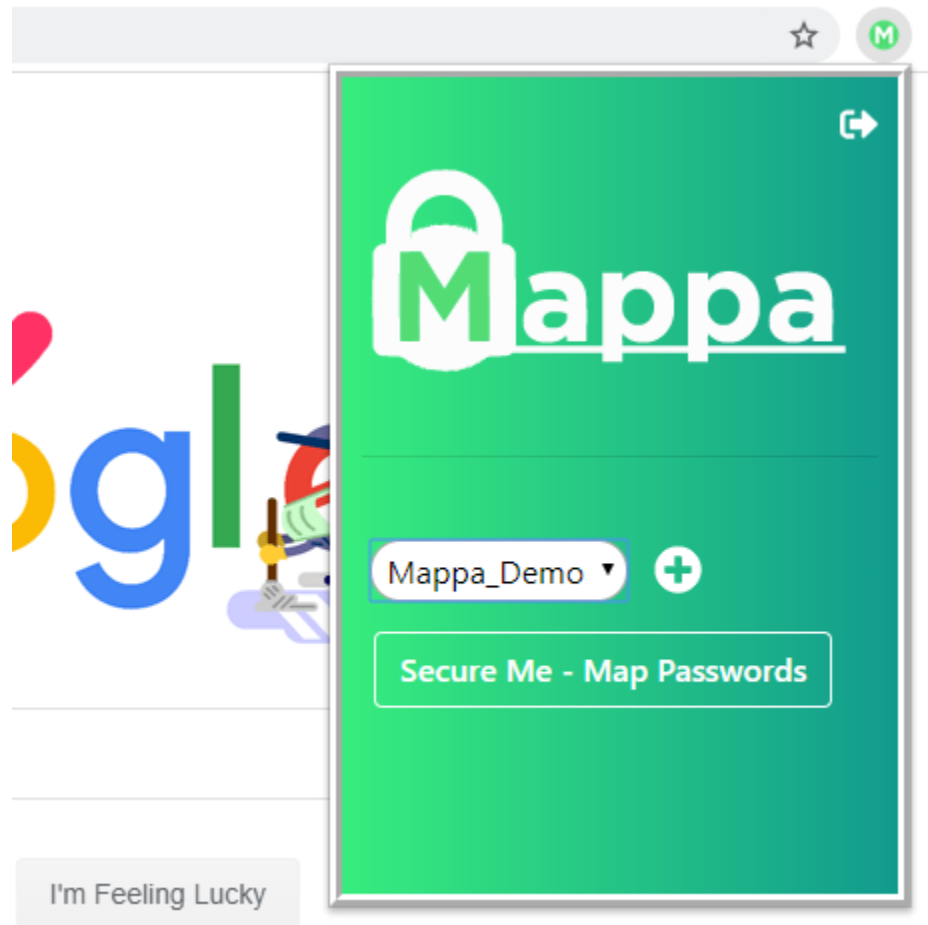
This page allows a user to select the map of the web service they wish to authenticate with and then press the button to map the password. A list of the user's maps is retrieved from the Web server using AJAX, a similar operation takes place on the click of the button, this time retrieving the map from the server. When the user clicks the plus icon, they are brough to the add website page.

Add Website Page

It is on this page that a user can create a new map for a service. For example, if the user wished to create a map for IT Carlow, they would enter "ITCarlow" into this box and click the button. The server receives the request and creates a new map for the user and associates it with the name specified.

This function is protected from Brute Forcing using map limits. The current limit is set at 50, meaning that a single user cannot have more than 50 maps.

Logout Function

As can be seen from each the screenshots of the extension, there is a logout icon at the top right-hand corner of each authenticated page. This button securely logs the user out.

General Use Flow

Once a user has set up a Mappa account and created account for their sites using Mappa. The process is as follows.

Navigate to desired site -> Enter the simple password the user remembers -> click the extension icon -> select the corresponding map
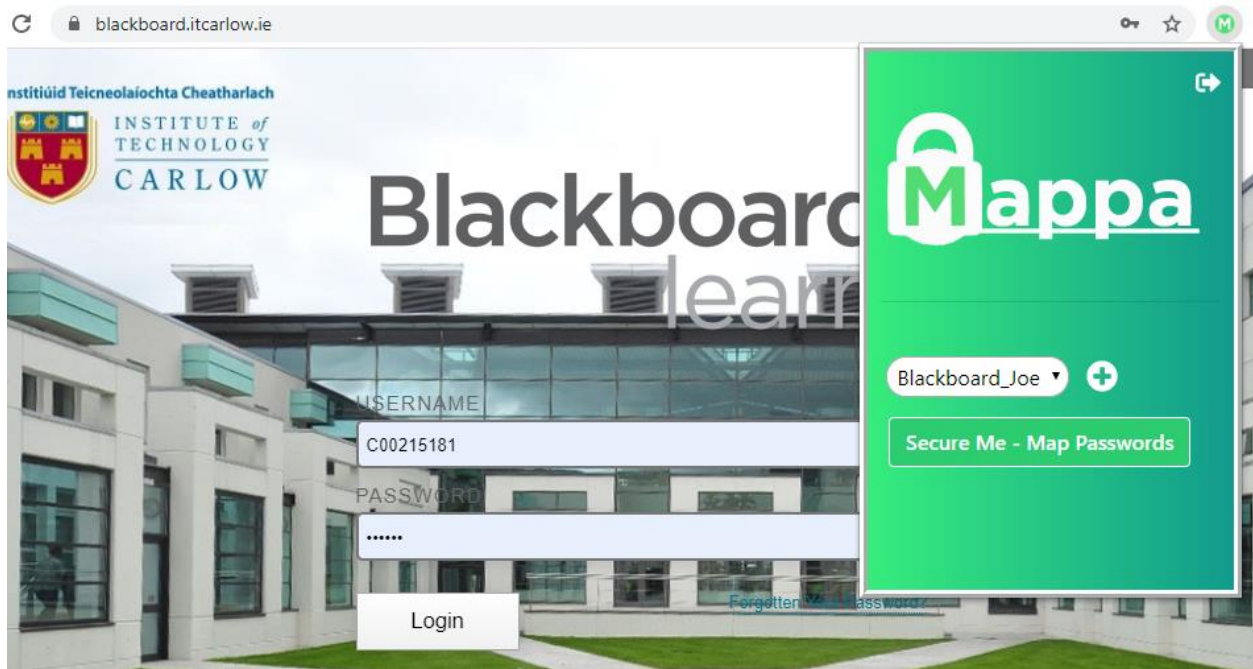
Then the user will press the Map Button, the value of the password box changes according to the map. The user can now log in as normal with the 3rd party service.

Development Challenges

Browser Extensions Requirements

Due to restrictions imposed on Browser Extensions, the code of the extension had to be changed numerous times. For example, inline scripts are not permitted in extension html code. While this in of itself is not a significant challenge, initial development of the extension interface was done using just html in a browser. Then when the interface was moved to the extension, it was not understood why the code was breaking, while this was a simple oversight, it did cost valuable research time. Additionally, due to Cross-Origin Resource Sharing (CORS) policy in modern browsers, the initial implementation of the browser extension had to be implemented differently. The initial implementation utilised CORS and the HTTP origin header to identify the Website the user was using Mappa on and select the appropriate map, however, because of federated services, where a single origin may be used to authenticate with separate accounts or services i.e. navigating to onedrive.com and signing into an IT Carlow account, this implementation could not work reliably and so control had to be given back to the user with the list box option, allowing them to manually select the map to use.

Extensive research was required to figure out how to inject content scripts onto a web page while also passing values to that script. The purpose of this being to pass the selected map into the DOM of the open page. This required the use of variables in Chrome local storage, utilising this required significant research of the Chrome and extension documentation.

In addition, because of the file structure of extensions, it was initially difficult to interpret how to use code found in support forums during the research phase.

Hosting

Amazon Web Services was used to host the project and so several processes had to be undertaken. A LAMP server was set up on an EC2 instance. A domain name was bought and DNS records configured appropriately, in addition headers were configured on the Web server, digital certificates obtained, Cryptographic suites configured for TLS and a Digital Certificate obtained for HTTPS. While all these would be relatively trivial tasks were this project to be repeated, considering this was the first ever time such infrastructure related tasks were performed by myself, the process consumed too much time.

Mobile Support

Significant effort was put into ensuring that the website for Mappa looks professional on both desktops and mobile devices. While this was achieved it was highly time consuming. Currently, the extension has not been tested on mobile devices due to Google Chrome for mobile currently not supported extensions. However, theoretically it should work, considering the extension is implemented in JavaScript using or implementing functions that are already available to Chrome for mobile.

Security

Extensive time was used to ensure the designed system is secure and safe to use from a Web Application vulnerability perspective. Security from common Web Vulnerabilities was

implemented in most cases from scratch and several scans were performed on the server. While this is a necessary step, it is heavily time consuming and can be resource intensive.
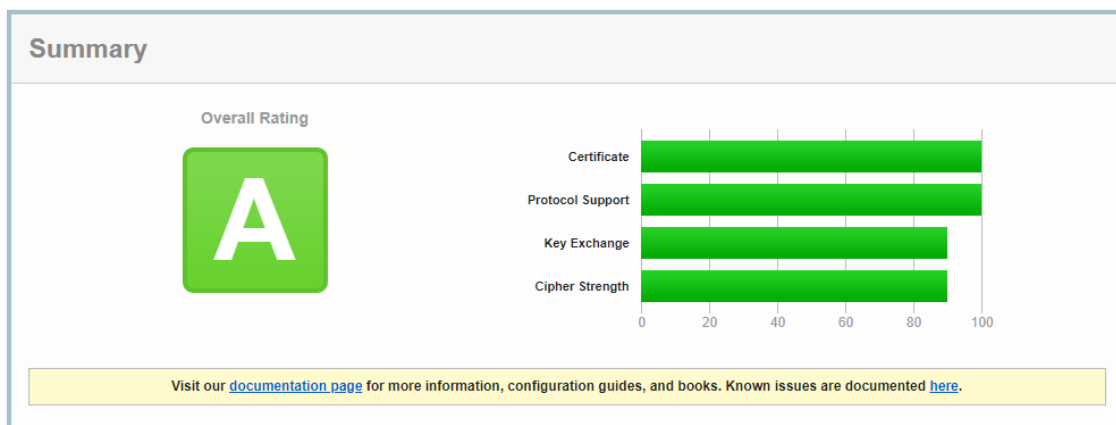
Testing

Security Testing

A penetration test was performed against both the web application and the web server running the application.

Nmap was used to scan for open/vulnerable ports on the Web Server, after the results of this test came back, unnecessary open ports were closed.

Nikto was one of the tools used to scan the server, this tool scans for numerous common vulnerabilities and misconfigurations. This tool was primarily used to ensure any header information given by the server abided by security best practices and was not an information disclosure risk. The results of this test prompted the hardening of the server through the apache config and removal of headers such as *x-powered-by,* which discloses the version of php on the server.

Qualys SSL labs web tool was used to ensure the security of the cryptographic suites used for HTTPS. After the results of this test returned a B, the supported cryptographic suites were changed on AWS and the current grade given by Qualys is A, as per the screenshot below.

ZAP proxy and Burp Suite were used to perform testing for common web vulnerabilities such as SQL inject and cross-site scripting, the application was not found to be vulnerable as it was designed with such web application vulnerabilities in mind. The first stage of this testing involved running a scan against the application and reviewing the findings. Any findings were successfully mitigated or found to be false positive, through the second stage of manually verifying whether or not findings were legitimate.

Finally, some manual penetration testing was performed on the server in an attempt to break the logic behind the application. While this particular component of security testing was not extensive due to time constraints, no medium, high or critical vulnerabilities were found.

User Testing

User testing was performed to assess the kind of emotion and behavioural responses user gave while using the application. Users that participated in testing signed up and used the application in the same way as described in the user manual, using the demonstration accounts, feedback received was taken on board. This user feedback influenced the final colour scheme and layout of the extension interface.

Usability Testing

From a usability perspective, the general using of the interface was tested, including edge cases, such as trying to create a map with no name, or a name too long to be processed by the server. For these reasons, conditions were enforced server-side to prevent these types of request crashing the server.

Additionally, a small group of people were briefed on the purpose of the application but now shown how to use it, and then asked to attempt to use the application, the icons for functions such as logging out and creating a new map were then updated to reflect this testing.

Comparison to Original Design and Specification

During development, Mappa changed significantly from the original suggested implementation and design. Initially, Mappa was going to be developed as a PowerShell module and once a password was mapped it would be copied to the clipboard. This was abandoned for two reasons, firstly for confining the users to Windows Desktops and secondly due to security concerns of having a user's password on the clipboard which can be read by other applications. This lead to the application being implemented as a browser extension as per the second researched implementation in the Research Manual.

As previously mentioned, the first implementation of the browser extension automatically identified the map to be used and automatically mapped the password. This version of Mappa was shown in the first status update presentation to project supervisors, it contained no user interface and mapped password inputs immediately on the click of the browser extension icon. This implementation was altered due to the issues of caused federated services and security concerns of working around the CORS policies of modern browsers.

Though the design of the final implementation changed significantly throughout the development process, all the specified functionality remained intact. Users can still successfully map passwords for any website that conforms to conventional password authentication, with the bonus of more granular control of the map used to transform a password. In addition, this browser extension can now work with several browsers (Chrome, Firefox) and on any desktop operating system. Theoretically, it should also work on mobile, should a later update of chrome support extensions.

The database schema underwent a very minor change late in development, an extra column was added in the security table to prevent the sign up and log in functions from starvation style brute forcing. An anti-CSRF token was linked to device identifiers.

The Map Generation function also underwent a late change in development, initially, generated maps mapped only alphanumeric characters to ensure compatibility with the most sites, as some sites do not accept certain special characters and what different sites accept varies widely. Special character support has now been added with the view to providing the user with options surrounding what kind of map to generate going forward, i.e. specify whether the map should contain special characters or not, in order to work with certain sites.

The interface for Mappa was also updated significantly during the project. Initially, there was a simple PowerShell interface, which became a lone browser icon that immediately mapped password inputs and finally became a fully fledged extension with a comprehensive user interface.

Learning Outcomes

The author has deemed to the following in first person.

Technical

During this project, several new technologies and languages were learned or understanding further expanded on. In some cases, such as with AWS the choices were made out of necessity, while some choice like the use of PHP as the server side language was deliberate as it simplified the initial configuration necessary to begin working on the project. Additionally, the use of JavaScript, AJAX, CSS and HTML was also out of necessity, due to web based nature of the project.

HTML and CSS

Using HTML and CSS to as required on the Mappa website enabled a professional looking website to be created due to already having a foundation level of understanding of these languages, however these topics had not been covered since 2nd year. Using these languages enabled the understanding of how elements and styles interact to be fleshed out and undoubtedly will prove to be a highly valuable skill going forward.

Bootstrap

Bootstrap was the CSS framework used to design the front-end elements of both the website and the extension interface. Bootstrap as a framework was completely new and provided a good opportunity to learn a new framework that is used widely across the web.

DNS

Buying a domain name and configuring DNS records was a very interesting and valuable part of this project experience, as although it is a relatively simple task, it is a highly valuable to get practical experience configuring it. This project also presented the opportunity to use various types of DNS records.

Amazon Web Services

Using AWS to deploy an instance in a Cloud environment is another highly valuable experience obtained and completely new. While underlying knowledge of what needed to be deployed was known, it was a completely new experience to do it.

Digital Certificates

Obtaining Digital Certificates merits similar value to the two technologies above, as while fundamentally a simple task, the experience of doing something like this for the first time is invaluable.

JSON

Prior to this project, I had never used JSON for anything before and did not know exactly how it was written. Though a simple data interchange format, it was valuable to apply it in a situation. JSON was used for the ajax responses sent throughout the application.

JavaScript

A much deeper understanding of JavaScript was obtaining during this project, particularly where html manipulation is involved. Indeed, the asynchronous nature was made apparent to me and, while an initial frustration, I learned to embrace it.

MySQL

Using MySQL on databases was last covered in the Year 2 module Web Development, it had been a long time since such operations were performed and this project was the perfect opportunity to hone those skills and get a greater sense of how the language is structured.

PHP

During this project, familiarity was gained with the syntax of PHP and available cryptographic libraries. The location and details of the php configuration files was learned and how to use correctly assign ownership of folders on the server to PHP.

Apache

Yet another technology that it was great to get hands-on experience with was Apache, the config files were used to harden the Web Server, particularly using headers sent with every response, configuring TLS and defining application root directories. While all covered through theory throughout the course, this was the first time applying the practical skills.

Technology Learning Outcomes Summary

While many of the technologies were previously known, I feel that applying them to the project with the view to making a secure, viable, successful and professional looking product significantly enhanced my understanding of the technologies and what they are capable of achieving when used together. One of the most valuable components of this project was combining small pieces of knowledge with technology capabilities only known in theory and learning to use them together to create the final product.

Personal

This section will outline personal learning outcomes gained during the development of Mappa.

Developing My Interests

Developing this project has forced me to learn my interests in the world of software development. One of the most interesting things I encountered was the use of AJAX requests. These requests while previously mentioned during the course of my studies were never covered in great detail, when combined with the use of JSON, which in itself falls into a similar category in terms of knowledge gained throughout my studies, became a very interesting topic for me. It was not until relatively late during the project development that I realised these requests could be used to query the server for information and thus control the view of the user interface.

JavaScript also falls into this category as initially I was dubious about using it so heavily for my project due to its reputation amongst programmer, I learned to enjoy writing in the language and became familiar with some of its quirks.

I thought I knew before the project that infrastructure configuration in terms of server requirements and load balancing is not something that I enjoy, while I acknowledge how necessary it is. After

working with these technologies, I learned a large amount about them and no longer feel that it is something that I would rule out working with in the future. As mentioned in the technical learning section, sometimes doing and applying theory for the first time can grow your mind and interests.

Throughout this project I tried to keep in mind the element of commercial viability and doing so has forced me to perceive the final product in a new light. I think this has also sparked

Time Management

Throughout my final year, a considerable amount of my time was split between college work, part-time work and familial duties. I found applying time optimisation strategies to be paramount to the success of this project. Optimising time in relation to this project involved making use of short periods of time, for example, utilising 20 minute waiting periods or lunch breaks to find the next article or tutorial video that I would watch that evening in order to be prepared for the practical coding elements. I learning that quite a large part of time is spent learning what tools, functions or libraries are available within a technology already saving tremendous time. Knowing what's available save creating functions and elements of the final product from scratch, learning what a language can do and if it fits requirements is of the utmost importance.

I found optimising small portions of time and allocating them to researching the possibilities of technologies to be the key to the success of this application while doing practical work during large free chunks of time, such as defining functions, it is a strategy that I am certain I will use again.

Achievement Review

Most of the development of Mappa was successful. As previously mentioned, the application achieves the function that it set out to achieve in enabling general user convenience without compromising their security. The application was brought to a stage where it is usable and presentable. Though the extension component was not uploaded to the Chrome Extension Web Store, it requires no extra no functionality for it to be placed there and made downloadable.

The below table of metrics was taken directly from the Functional Specification and each goal is categorised as successful or unsuccessful.

Successfully Achieved

Criteria	Description
Security	All User Data including character maps are secure
Security	The Web Application is not vulnerable to SQL
Security	The Web Application is resistant to brute forcing particularly on the sign up and sign in pages
Security	The Web Application is not vulnerable to XSS
Security	There is appropriate handling on sessions by the Web Application
Security	The Web Application is not vulnerable to Cross-site Request forgery
Security	The Web Application is not vulnerable to OWASP top 10 and other common Web Vulnerabilities
Security	The Web Server is hardened and uses modern and secure Cryptographic Suites for TLS
Security	A valid digital certificate has been obtained and is in use on the web server
Security	A penetration test was carried out on the system with no high/medium severity issues found identified. Any issues found were successfully mitigated.
Platform	The system works on common Web Browsers Google Chrome and Firefox, independent of operating system
Password Strength	The correct use of this system enhances password strength
Usability	A user can install the system quickly
Usability	The system loads and maps the user's text quickly
Usability	The system is simple, intuitive to use and presented well

Usability	The browser extension correctly identifies password fields and maps the text within them according to the user selected map
Usability	Users can login from both the web site and on using the browser extension
Usability	Users are prompted appropriately at all stages of user input
Reliability	The Web Application is hosted on the internet and works with correctly with the Browser Extension
Reliability	The system works as designed and does not throw exceptions during standard operation conditions
Support	There is adequate documentation explaining the inner workings of the system and how to use it. An easy to follow instructional video is available on the Web Site.

Not Achieved

Criteria	Explanation
Mobile Support	The final iteration of the project does not support mobile usage, while theoretically the concept would work, Google Chrome for mobile does not support browser extensions.
Extension uploaded to the Web Extension Store	The Web Extension was not uploaded to the Chrome Web Store due to time constraints, however there is no extra coding required to upload it.
Account options functionality from within the extension	Ideally, all the functionality of the application should be possible from the extension, including sign up, password change and similar functions. However this was not possible in the timeframe.
Map Redundancy	An option for the user to download their maps was not completed due to time constraints.

Future Versions

While much was achieved during the project, a future iteration could quite possibly implement the functionality within the Not Achieved section. Additionally, this application could also be implemented as a script that could be plugged into other websites. Such as by using the *<script src>* functionality. This would be used by 3rd parties in the same way that chatbots are pulled into support website and could enable Mappa to be pulled into the trusted application to provide

security. This would be a useful feature within mobile applications that are simply an interface to their corresponding web application counterparts.

Additionally, some websites have varying password policies, making it more difficult to use the current iteration of Mappa with them, if the map generated uses too many characters or characters not supported by the site. A future iteration would give the user slightly more control over the map generated without compromising the security that comes from the fact that it is generated randomly.

There are always more features that could be added to this application as well as different implementations, expanding the potential target market and platforms it can run on. This app could benefit from integration with other services such as Joomla, a content delivery service. The usability aspect of creating accounts with Mappa for the first time could also be examined and improved upon. Overall, the final product is acceptable due to its functional abilities and minimalistic interface.

Plagiarism Declaration

I declare that all material in this submission is entirely my own work except where duly acknowledged. I have cited the sources of all quotations, paraphrases, summaries of information, tables, diagrams or other material; including software and other electronic media in which intellectual property rights may reside. I have provided a complete bibliography of all works and sources used in the preparation of this submission. I understand that failure to comply with the Institute's regulations governing plagiarism constitute a serious offence.

Student Name: Kevin Davitt

Student Number: C00215181

Signature: _____

Date: _____