ASHLEY JAMES DOWIE

sit 120 aSSEMENT 1: Proposal and proof of concept

Table of Contents

[Product Summary 2](#_Toc79169009)

[Asset list 2](#_Toc79169010)

[Product Purpose: 5](#_Toc79169011)

[Target Audience 5](#_Toc79169012)

[Individuals / personal use: 6](#_Toc79169013)

[Organisations / Business: 6](#_Toc79169014)

[User stories 6](#_Toc79169015)

[Why Password Place is creative: 8](#_Toc79169016)

[UX/UI Design 9](#_Toc79169017)

[Major Components and their intended behaviour: 11](#_Toc79169018)

[Guest mode password generation form 11](#_Toc79169019)

[Member mode password generation form 11](#_Toc79169020)

[System summary 15](#_Toc79169021)

# Product Summary

Password Place is a product which allows users to generate passwords for various applications and websites. Password place requires the user to only remember a single password, their ‘primary password”. Password Place uses a deterministic function to create unique password for each application based on the name of the applications name and the user’s primary password. Password Place does not store any user passwords, they are always generated on the fly with the deterministic function.

Password place offers two options for use. The user can choose to use Password Place without creating an account. Or the user can choose to create an account. If the user chooses to use Password Place without creating an account, they will be able to get a unique password for any application simply by typing in a password and a websites name. As the function for determining the password is deterministic, the password will be the same for the user each time.

Users that choose to create an account will have access to more advanced features. A user that signs in will be able to create new application entries and save these to a list. This means the user does not have to try and remember which applications they are using Password Place generated passwords for. Users will also be able to “step forward” and “step back” passwords. This allows passwords to be generated using a third parameter, which is a simple integer value known as “step”. Step forward allows a user to generate a new password for an application in the situation that the application has password expiration after a set time. Step back allows a user to go through the history of their passwords and recall previous passwords.

For a logged in user the only information that is stored is their sign in password, their list of applications and the current step for the application. Users are recommended to not use a primary password that is the same as their sign in password. By not storing the user’s primary password, even if someone gains access to the user’s account, they will not be able to determine any of the user’s passwords.

# Asset list

|  |  |
| --- | --- |
| style.css | The main styling sheet to be used throughout the website. This sheet uses media queries to create break points to make pages as responsive as possible. The main page layout is determined using grid areas. Other features provided by style.css include tooltips, a navigation bar and a self-contained content scrolling pane. |
| index.html | The main html page that will be used to display all the Vue templates for the application. As the first draft of this product contains no Vue files, index.HTML contains all the DOM elements directly. Index.html makes use of various Vue directives, using the Vue component within index.js called “lister”. Index.html currently only contains a demonstration for the “Member” section of the product. Some elements in index.html make use of inline CSS. |
| main.js | This JavaScript file will eventually contain the required Vue imports and router. However, as the first draft for this product contains no Vue files, main.js currently contains all of the below Vue components mentioned below. |
| OptionButton.Vue | This Vue file will contain the Vue component called “OptionButton”. Option buttons will appear in a list at the bottom of member’s page. Each option button represents an application that the user has generated a password for. Each option button has a prop called “name”. The name prop is used within the template of the option button. The option button’s template displays a tooltip when hovered over and contains a button that the user can select. Upon clicking an option button, the option button component will emit up the chain to the root Vue component “Lister”, updating the currently selected application. The currently selected option button will also update to reflect changes to the applications name from the main form component. |
| PasswordGenerator.Vue | This Vue file will contain the Vue component called “PasswordGenerator”. The password generator component generates a password for the currently selected application and displays it. Password generator has 3 props, “AppName”, “PrimaryPassword” and “Step”. These values are used by the computed field “Generate password” which is in turn used in the template for password generator to display the generated password for currently selected application. The prop values “AppName” and “PrimaryPassword” are determined by the main form component, whilst the step value is determined by the stepper component. |
| Stepper.Vue | This Vue file will contain the Vue component called “Stepper”. The stepper has one property called “Step”. The stepper component provides a UI for the user to adjust the step value. The stepper component displays two buttons with the current step value displayed between them. The user can increment the current step by clicking the button to the right or decrement the step by selecting the button to the left. The step value will update to reflect the changes from the user. Whenever either of these buttons is pushed, the stepper emits up the chain to the lister component, causing the list to update the step value. The step value is used by the password generator to update the password of the currently selected application. The user can decrement the step value to see passwords that they have previously used for this application or increment the step to update the password in case of password expiration. |
| MainForm.Vue | This Vue file will contain the Vue component called “MainForm”. Maniform has two props “AppName” and “PrimaryPassword”. The template of the main form provides two input fields for both the application name and the primary password. As application name is specific to the currently selected application, the application name field will change to reflect the user selecting an option button component representing a different application. Likewise updating the application name from the main form will update the name of the option button representing the currently selected application. Primary password is not specific to an application, so this value will only change when the user updates this using the input field in the main form. The primary password and application name entered into the main form are used by the password generator component to generate a password for the currently selected application. Whenever the user makes a change to the primary password or application name input fields, main form emits up the chain to the lister component. This makes lister update the relevant data to reflect the change. |
| Guest.Vue | This Vue file will contain the Vue component called “Guest”. The purpose of the guest component is to render the guest page. The guest component has the props “AppName” and “PrimaryPassword”. The guest component is passed these props from the root Vue component “Lister”. Guest contains the main form component and password generator within its template. The guest component always passes the password generator component 0 for the prop “Step” as stepping is a member’s only feature. The guest component also does not include option buttons this is also a members only feature. Like the member component, the guest component acts as an intermediary between all other components and the lister component. Password generator and main form emit values to guest which in turn emits values to lister. Lister passes props to guest, which in turn passes props to password generator and main form. |
| Member.Vue | This Vue file will contain the Vue component called “Member”. The purpose of the member component is to render the member page. The guest component has the props “AppName”, “Step” and “PrimaryPassword”. The member component is passed these props from the root Vue component “Lister”. Member contains the main form, password generator, option button list and stepper within its template. Like the guest component, the member component acts as an intermediary between all other components and the lister component. Password generator, main form, option buttons and stepper all emit values to member which in turn emits values to lister. Lister passes props to member, which in turn passes props to password generator, main form, option buttons and stepper. |
| Lister.Vue | Lister.Vue is the main root component. Lister contains all the data used throughout Password Place. Lister passes the props “AppName”, “PrimaryPassword” and “Step” to the member component. Lister passes these props to the guest component as well, apart from “Step”. Member and guest emit all these properties back to lister as well. Lister contains all the methods that provide the appropriate response to emits being passed up to it from member and guest. |

# Product Purpose:

Password Place was created to offer users the same level of password protection they would have if they used a unique password for every application. But to only require them to remember a single password.

# Target Audience

Password place targets any user who wants strong password security but want the convenience of not needing to memorise a unique password for every application they use. This product could appear to larger organisations or individuals alike.

Password place can be used for generating passwords for entertainment purposes such as Netflix or for business purposes such as generating passwords for products such as MYOB. The only limitation of Password Place is that passwords are determined by the application, so users would need to be willing to change their existing passwords to make use of the app. This could also potentially not suit some purposes such as an organisations security codes that may be required to be predetermined by the entity.

Individuals / personal use:  
Many individuals use a single password for every application they use. This causes two problems main problems for the user. These problems are inconvenience and security risk.

The first problem is that if an application has a password expiration policy, the user is going to need to change their password for that website. The user now either has the choice of remembering an additional password each time this happens, or they will need to update their password for every application they use. Password Place offers a convenient solution for the user as they only must remember a single password. If a password for an app expires, the user simply needs to “step forward” their current password for that application and then update their password on that application only.

The second problem is that if the user uses the same password for every application, someone who discovers that password will have access to every application that the user uses. Password Place fixes this problem by ensuring that the user will have a unique password for every application that they use. If someone were to discover one of the user’s passwords, they would only have access to the one application associated with that password. Password Place does not send the user’s primary password to the server side, all processing occurs on the client side. This greatly reduces the likelihood that the user’s primary password will ever be stolen.

## Organisations / Business:

Organisations generally use a unique password for each application that they use. However, to keep track of these password they are normally stored. As the systems of larger organisations are often the target of cyber security threats, storing passwords creates a large security risk for organisations. This cannot be resolved by expecting employees to memorise a large list of unique passwords for the organisation as the number of applications an organisation uses would be prohibitive. It could be possible to have employees memorise passwords for the applications that are only immediately a part of their role, however this would create knowledge silos that pose a risk of the password being lost if the staff member were to leave the organisation. Additionally, if passwords were updated, it would be difficult to share the new password changes across the organisation without using means that pose a security threat, such as emailing the updated password.

Password Place removes the requirement and the risk of the organisation storing passwords. Organisations also do not need to be concerned about their passwords being compromised if password place were to succumb to a cyber security attack, as Password Place does not store passwords. Password Place dynamically generates passwords with a deterministic function. The only information that Password Place stores is the user’s log in password (which is not the same as their primary password), log in name and their list of applications, each having an application name and step count.

## User stories

* Russell is a small business owner within the IT industry. Russell’s business uses a lot of third-party services and applications to help meet the needs of his business as his business is too small to try and do everything in house. Russell’s business had previously been storing their passwords in an excel spreadsheet for all the third-party applications they use.   
    
  Unfortunately, Russell’s business became the victim of a cyber security attack and the passwords within the excel spreadsheet were compromised. Russell has considered having a third party with better security practices and infrastructure store his businesses passwords, however Russell is unsure if he wants to trust all his passwords with a third party. Russell would like to find a way to meet his need for maintaining many passwords without the risk that comes with storing them.
* Jenny is very active on social media and holds numerous accounts on various platforms. Jenny uses a single password for all her social media accounts. Jenny does this as it is easier to remember one password and she doesn’t want to risk forgetting a password and being locked out of one of her accounts  
    
  One day on of Jenny’s friends watched her typing in her password. Her friend later went into all of Jenny’s social media accounts and left embarrassing messages there. Jenny now realises that if she had used a unique password for every account, only one of her accounts would have been compromised. However, Jenny is not sure she can remember many passwords off the top of her head. Jenny wishes that she would only have to remember a single password and not have to worry about all her accounts being compromised again.
* John uses a lot of different applications for various things. John is conscious of internet security, so he uses several passwords for his different apps. John has all his passwords memorised and this has not caused any problems for him so far as he is confident that he can remember them all as they are ingrained in his brain. One day one of John’s passwords expires. John is in a hurry to do what he needs to on the application quickly so he can head off to work. John quickly enters a new password and then leaves for work.  
    
  A week later John decides to use the application again. John goes to enter his password and it is denied. John remembers that he changed the password last week, but he cannot remember what the password is. John is now locked out of the application. John wants a way to track the history of changes to his passwords.
* Merry owns multiple different devices, often alternating between them, using her tablet, mobile phone, pc and laptop. Because of this, Merry prefers to use web-based applications that do not require installation.   
    
  Merry is very conscious of password security as she is an advanced user of technology and knows the risks of and potential ramifications of security breaches. Merry is interested in finding a password generation software, however she would prefer the software to not require installation on any of her devices. As Merry is very security conscious, she is very reluctant to share any of her personal information online.   
    
  Merry has browsed through many different password generation software options. All the software generators she has come across have required the user to either create an account or install the software on their device, often both. Merry is still searching for a password generator that does not require the user to install any software on their devices or create an account.
* Glen is elderly and not very confident with technology. Covid-19 restrictions and lockdowns have forced Glen to become more active in the use of technology. Without the use of social media and emails, Glen would not be able to keep in contact with family and friends. Glen has also been forced to use internet banking due to not being able to visit the branch in person anymore.  
    
  As a person who is not comfortable with technology, Glen is looking for ways to make the transition as easy as possible. One of the problems Glen has is that Glen is only able to remember a simple password. However, some of websites that Glen uses require a password of a great length with combinations of letters, numbers, and special characters. Glen is somewhat nervous and confused when it comes to entering special characters with the keyboard, as these often require the user to hold shift when entering.  
    
  Glen would prefer to just use a simple password that is easy to remember and type. However, Glen must comply with the password requirements of these websites in order to use them. Glen wishes there was a way that he could keep using these websites, but only need to type in and remember a simple password.

# Why Password Place is creative:

Password Place is creative because it offers a solution to the three main concerns of password security. These three main concerns are external risks (passwords being compromised by a third party, internal risks (the user forgetting a password or losing the passwords they have stored) and convenience.

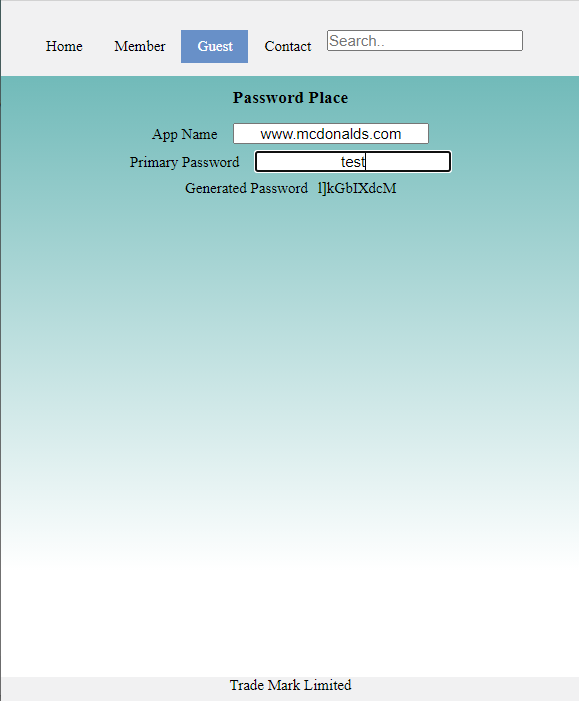
The use of a deterministic function greatly reduces external security threats. Passwords are not stored in neither Password Place nor on the user’s end. This means that if someone were to look through all the storage of either system, no passwords could be obtained. When Password Place generates a user’s password, this happens on the client side. This means that a user’s primary password will not be intercepted as it could be if it were sent to the server and the user’s generated password cannot be intercepted either.

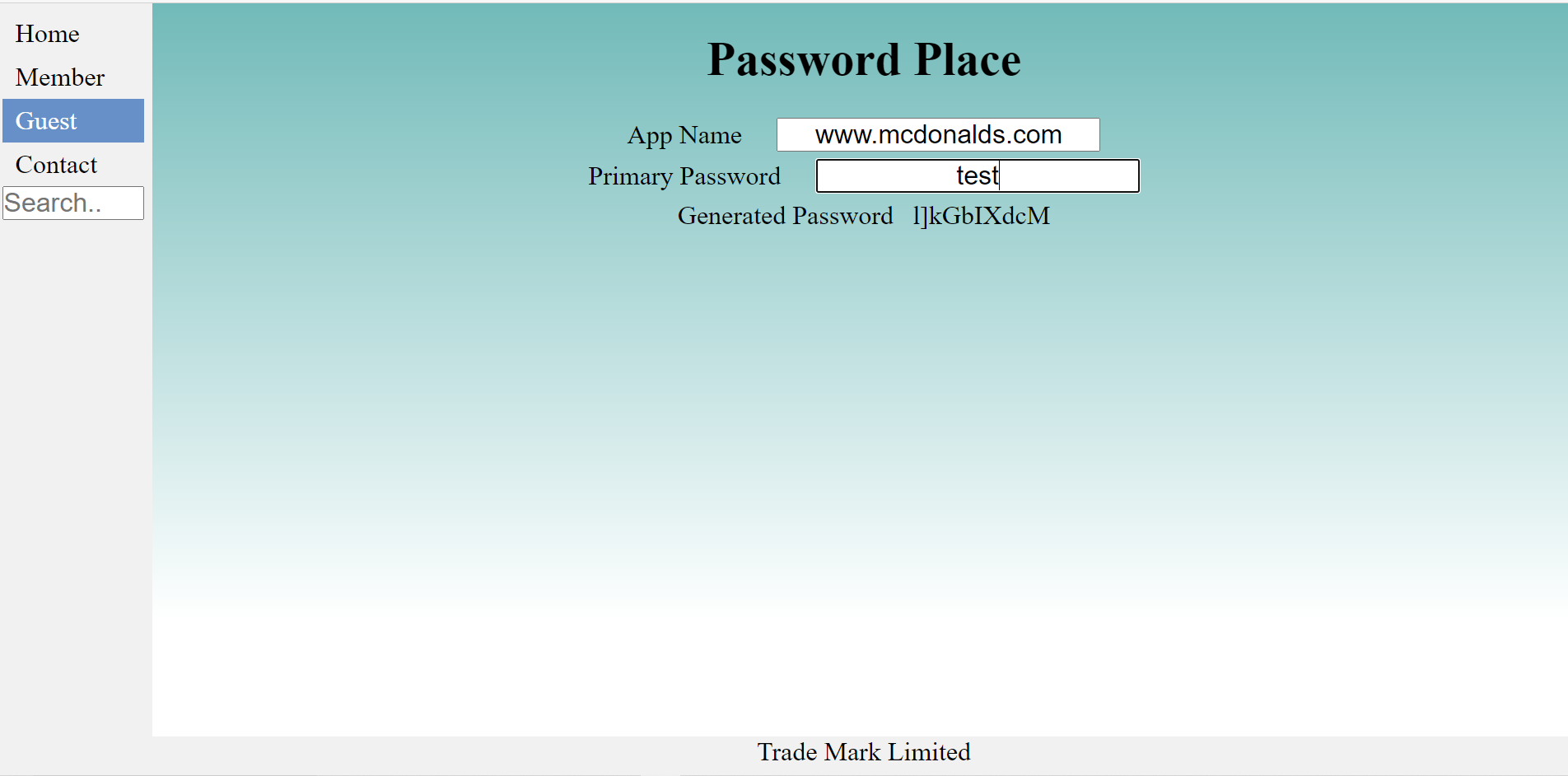
By using a deterministic function to generate passwords, passwords cannot be lost. Password Place’s step feature allows users to maintain a complete record of their previous passwords. This means this even if a user needed to remember a previously used password, they can retrieve it at any time. A user cannot lose their password. So long as a user remembers their login name or their login password and their primary password, their password cannot be lost.

Password place offers great convenience to the user. The user will no longer have to remember a password for every application, yet they will gain the same benefit as they would if they did. Complex passwords can be generated even if the user’s primary password is relatively simple. Even if a password were to expire, the user can generate a new password for the app with a single click of the step button.

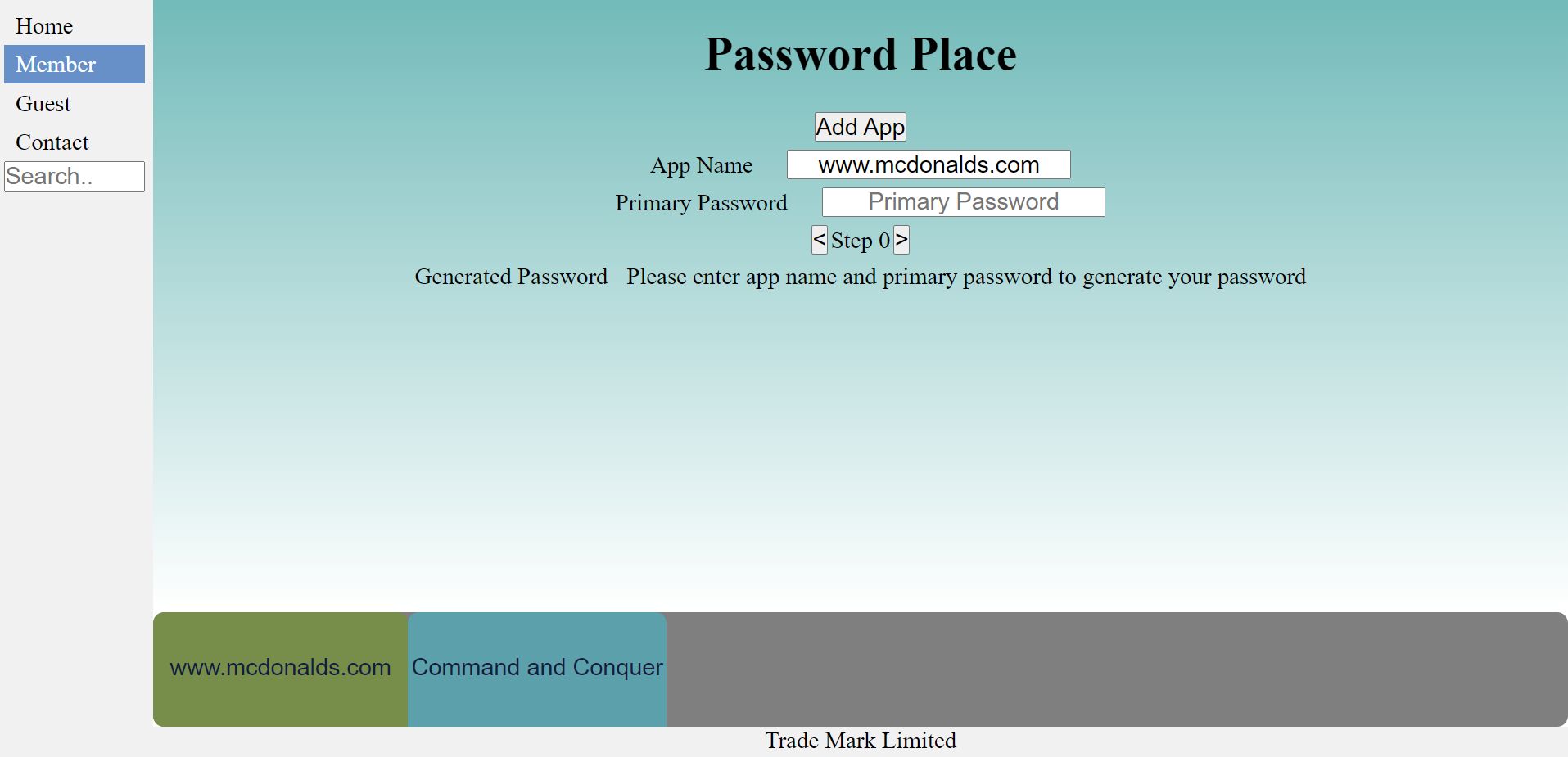
# UX/UI Design

Password Place is a responsive application and will attempt to optimise the layout and size of page elements to best display on different screen resolutions. The navigation bar switches between being displayed at the top or the left side of the screen based on screen width. The navigation bar will also swap between being vertical or horizontal based depending on screen width. As the screen width shrinks, Password Place attempts to make greater use of vertical space to make more room horizontally. Below is an example of the guest page of Password Place. The first image shows Password Place displayed at a mobile resolution. The image shows Password Place displayed in desktop resolution.

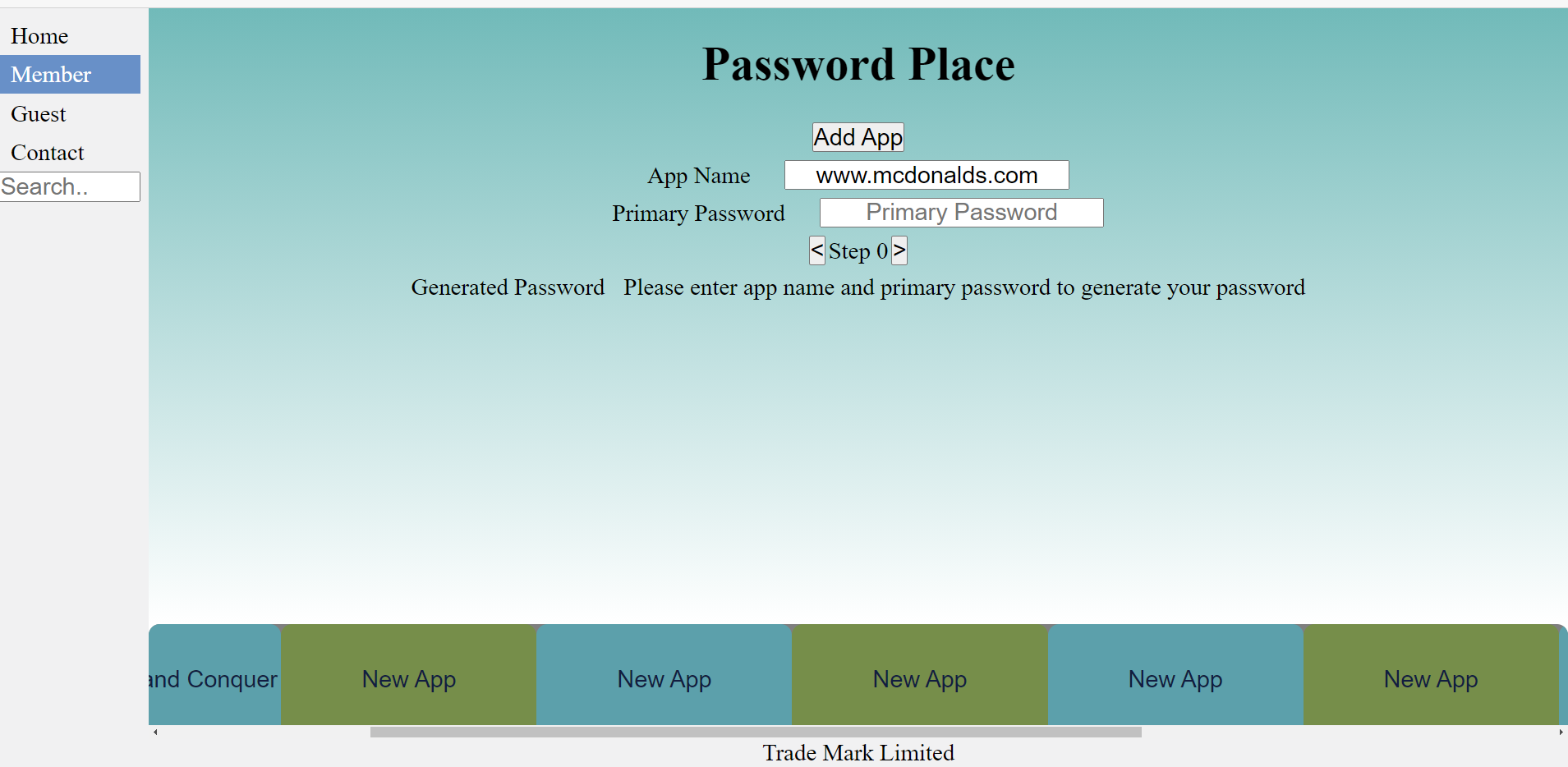




Below is an image of the Members section of the website. At the bottom you will notice a list of apps that the user has added.



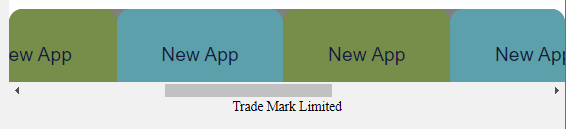
If the user were to add enough apps that the buttons could not display within the lower pane, a scroll bar will become visible. The user will then be able to scroll through the list of items, without scrolling the other sections of the page.



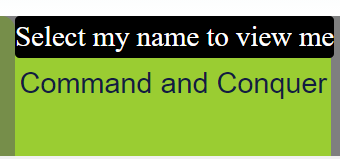
As the width of the page decreases, Password Place will reduce the amount of apps that it attempts to display.







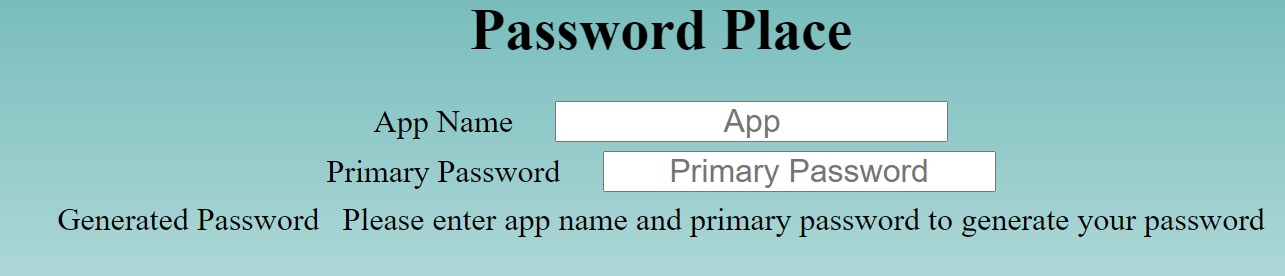
Password Place makes use of tool tips to help guide and instruct the user.

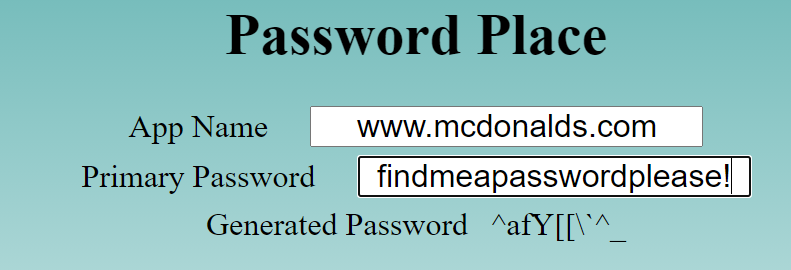


# Major Components and their intended behaviour:

## Guest mode password generation form

Below is an example of the guest mode password generation form. The form is very simple to use. To generate a password, Password Place needs two things, an application name and a primary password. The user simply needs to enter the name of the application and then their primary password. Once they have done this, a unique password will instantly be created for them.



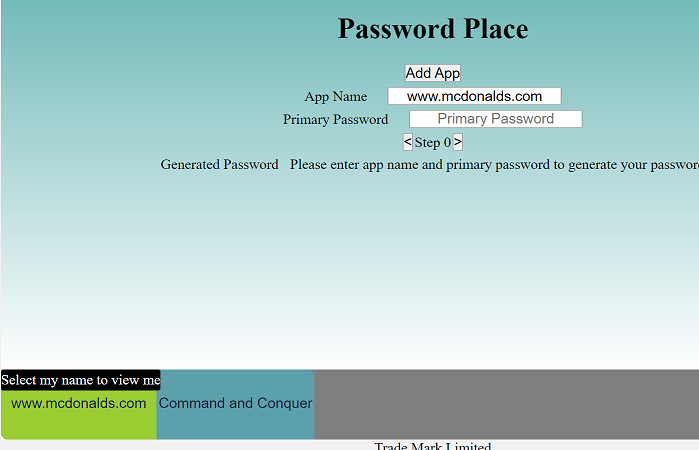


## Member mode password generation form

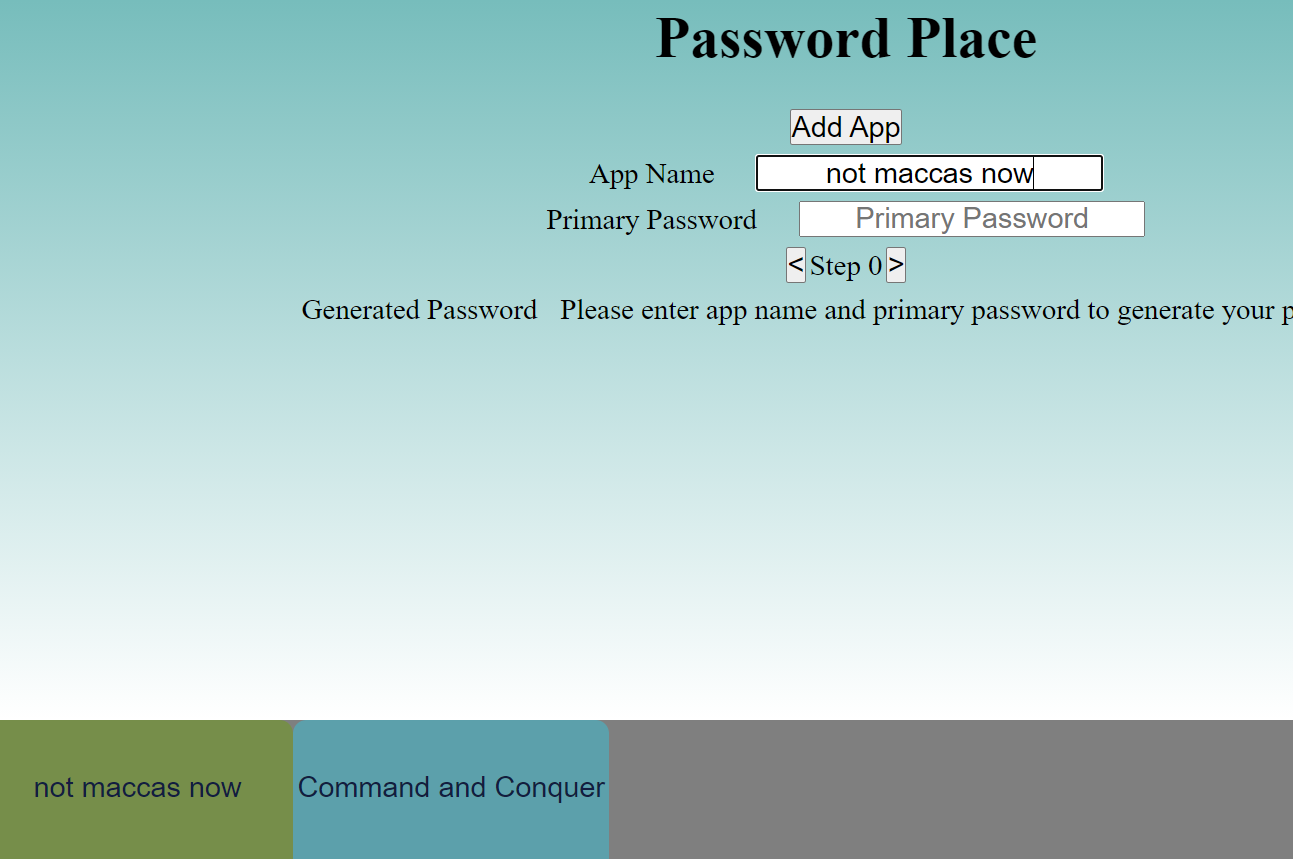
The Member mode password generation form offers the same base functionality as the Guest mode password generation form with a few additions. Firstly, the user is able to maintain a persistent list of all of the apps that they have generated a password for and the current “step” of that app. Users can add more apps to their list by simply select the “Add App” button. Users can also select existing apps from the list to generate a password for that application. The user can edit the information of any app that they currently have open.

Below the user is currently on the “Command and Conquer” App. All they need to do is select [www.mcdonalds.com](http://www.mcdonalds.com) from the list to select that app:  

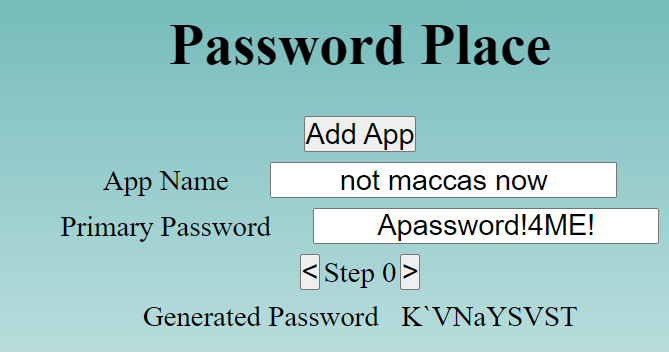

The password information will not be relevant to www.mcdonalds.com



The user can easily edit the name of the selected app simply by typing in the app name input section and the apps name will instantly change to reflect this on the list.



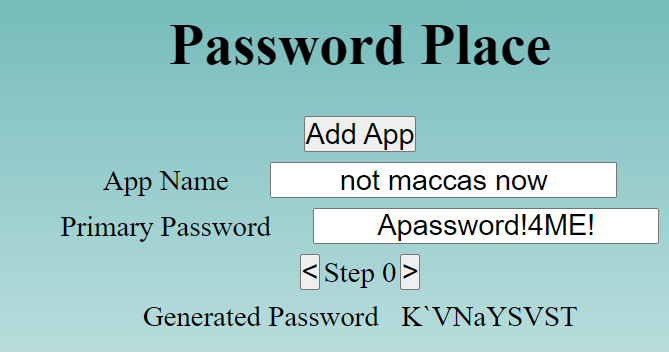
The other major feature that the Members Mode form offers is the ability to increment or decrement the apps “step” counter. The user does this simply by selecting the arrow keys next to the step value. This allows passwords to be modified in case they have expired. The user is also able to step back to see their previous password such as below:

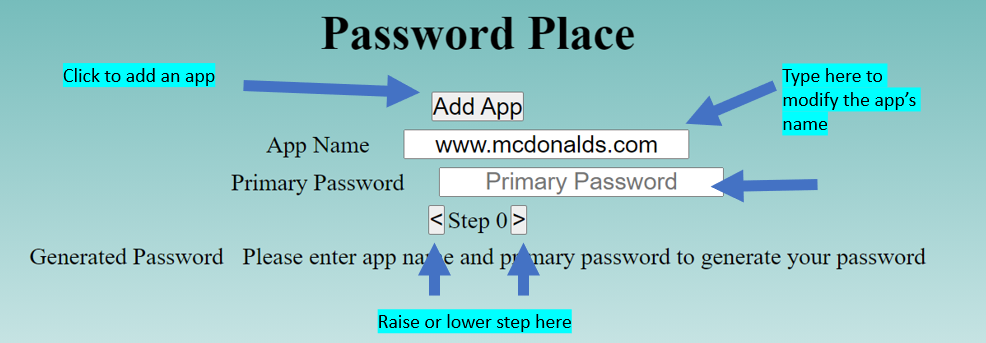


Step forward:



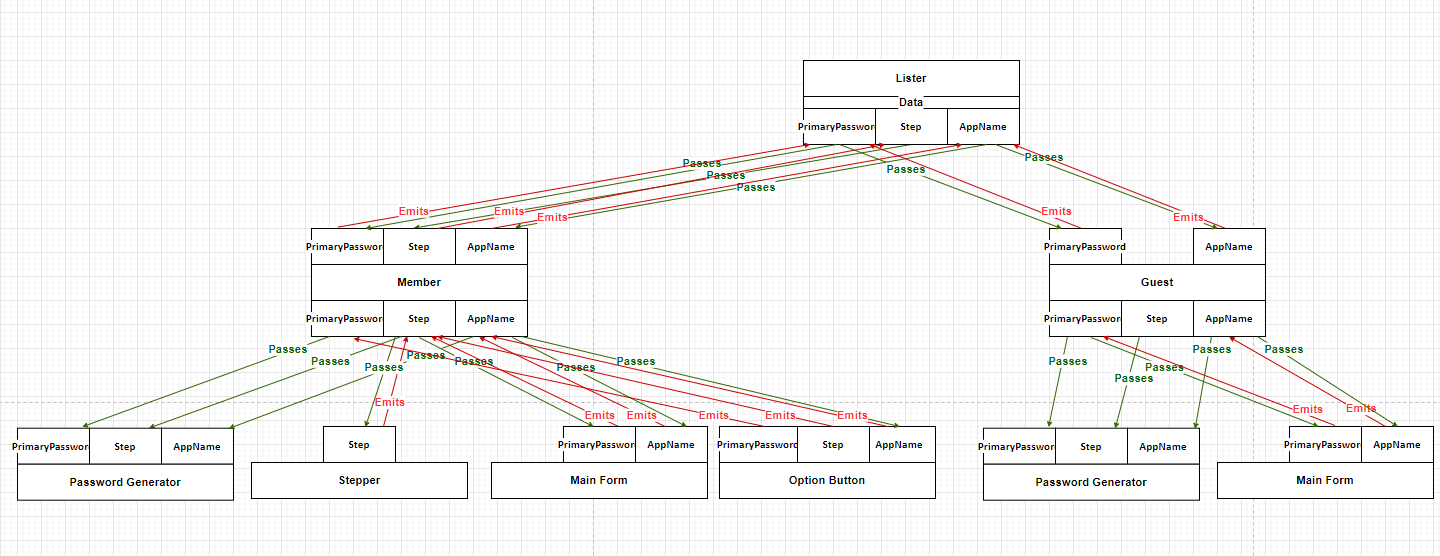
Step back:





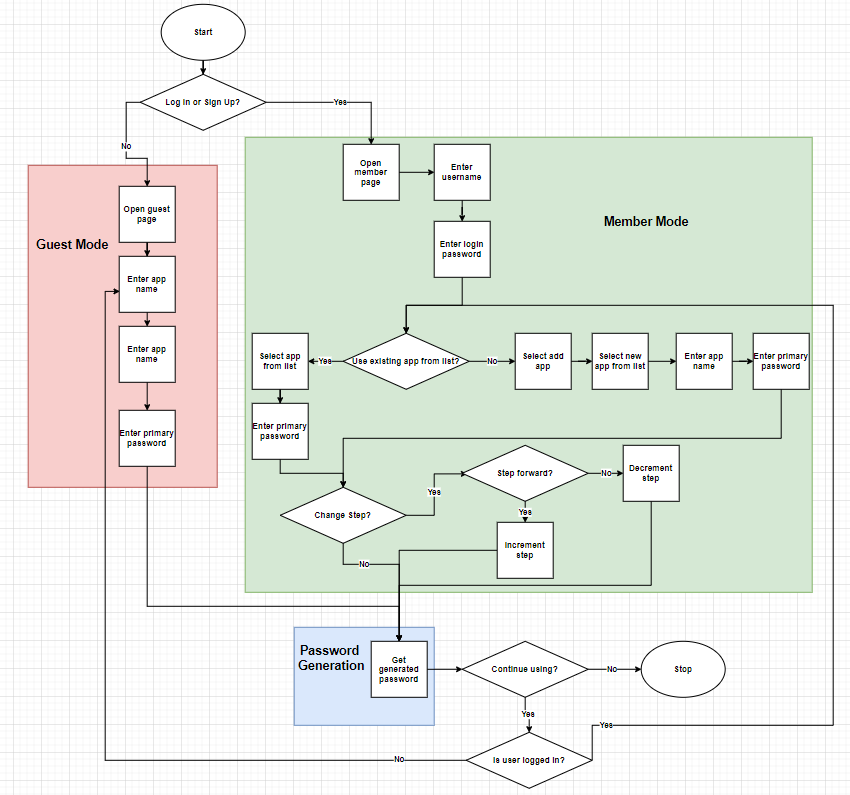


Once all the required information is entered. The generated password will be instantly displayed.

Below is an example of Vue components used in Password State and how they will interact with one another.  


# System summary

Password place gives users the choice of two modes. Users have the choice of either using guest mode or member mode. Guest mode does not require the user to log in. However, the drawback is that the user will not have access to the step feature and will not be able to save apps to a list. Member mode allows users to save apps to a list, removing the need for them to have to enter the name of the app they wish to generate a password for each time. Member mode also allows users to make use of the step feature



All files can be found at the following GitHub link:

https://github.com/AshleyDowie/SIT120---Introduction-To-Responsive-Web-Apps/tree/main/assignment%201/SIT120210694097A1-ProposalAndProofOfConcept