

Home Page

Home

Following

Events

Account

Post

[Sidebar is fixed
size ~20% width of
the screen]

Login/Signup

[Is a logout button
instead when
signed in]

Account Name [Edit]

[IMAGE]

-----Text-----

[Un/follow button]

Edit button only appears
for users that made the
post, or system admins
Brings you to post page
with details autofilled

Following an organisation
is the same as “joining”
them, which causes
posts to appear in the
“following” page

Simply shows the latest
posts, whether the user
is signed in or out

Clicking follow while
signed out redirects to
login page

Account Name [Edit]

[IMAGE]

-----Text-----

[Un/follow button]

Main page content all stretches to screen size, ~80% of screen width

Screen height of posts should stretch down to fit screen width

Clicking Logout on any
page re-directs you to
this page (home page)

Following Page

Home

Following

Events

Account

Post

[Sidebar is fixed
size ~20% width of
the screen]

Logout

[Is a Login/Signup
button instead
when signed out]

Account Name [Edit]

[IMAGE]

-----Text-----

[Un/follow button]

Edit button only appears
for users that made the
post, or system admins
Brings you to post page
with details autofilled

If an organisation is
unfollowed, posts don't
disappear unless the
page is refreshed.

Accessing this page
while signed out simply
redirects the user to the
login page

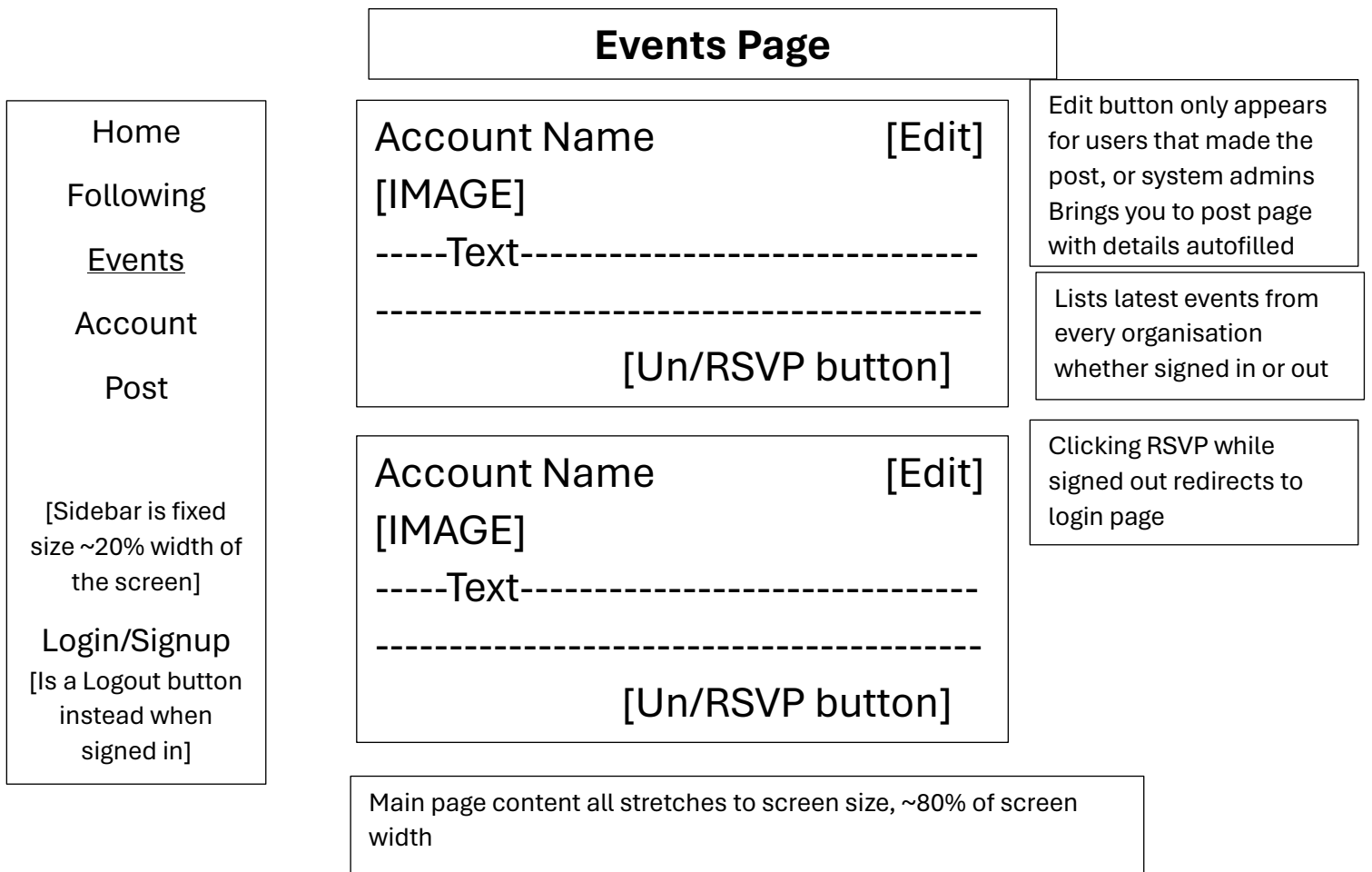
Account Name [Edit]

[IMAGE]

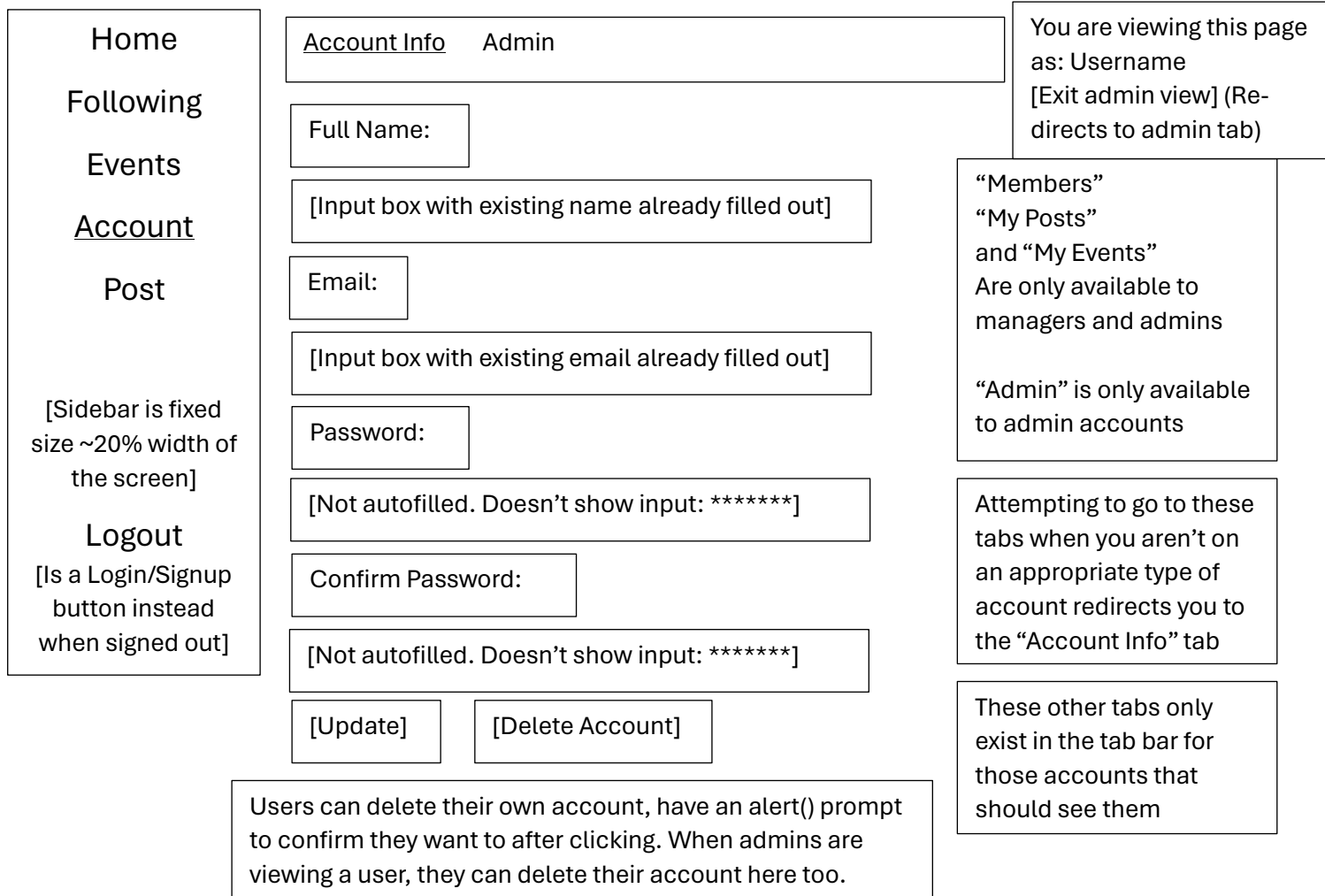
-----Text-----

[Un/follow button]

Main page content all stretches to screen size, ~80% of screen
width



Account Page - User



Account Page (Branch) - Members

Account Info Members My Posts My Events Admin

You are viewing
this page as:
Username
[Exit admin view]
(Re-directs to
admin tab)

Home

Following

Events

Account

Post

[Sidebar is fixed
size ~20% width of
the screen]

Logout

[Is a Login/Signup
button instead
when signed out]

Name

Name

Name

Name

Name

...

Account Page (Branch) - My Posts/Events

Home

Following

Events

Account

Post

[Sidebar is fixed
size ~20% width of
the screen]

Logout

[Is a Login/Signup
button instead
when signed out]

Account Info Members My Posts My Events Admin

Account Name [Edit]

[IMAGE]

-----Text-----

[See RSVPed]

Account Name [Edit]

[IMAGE]

-----Text-----

[See RSVPed]

(Re-directs to
admin tab)
You are viewing
this page as:
Username
[Exit admin view]

[List of names who
have RSVPed to
the last clicked-on
event]

Name

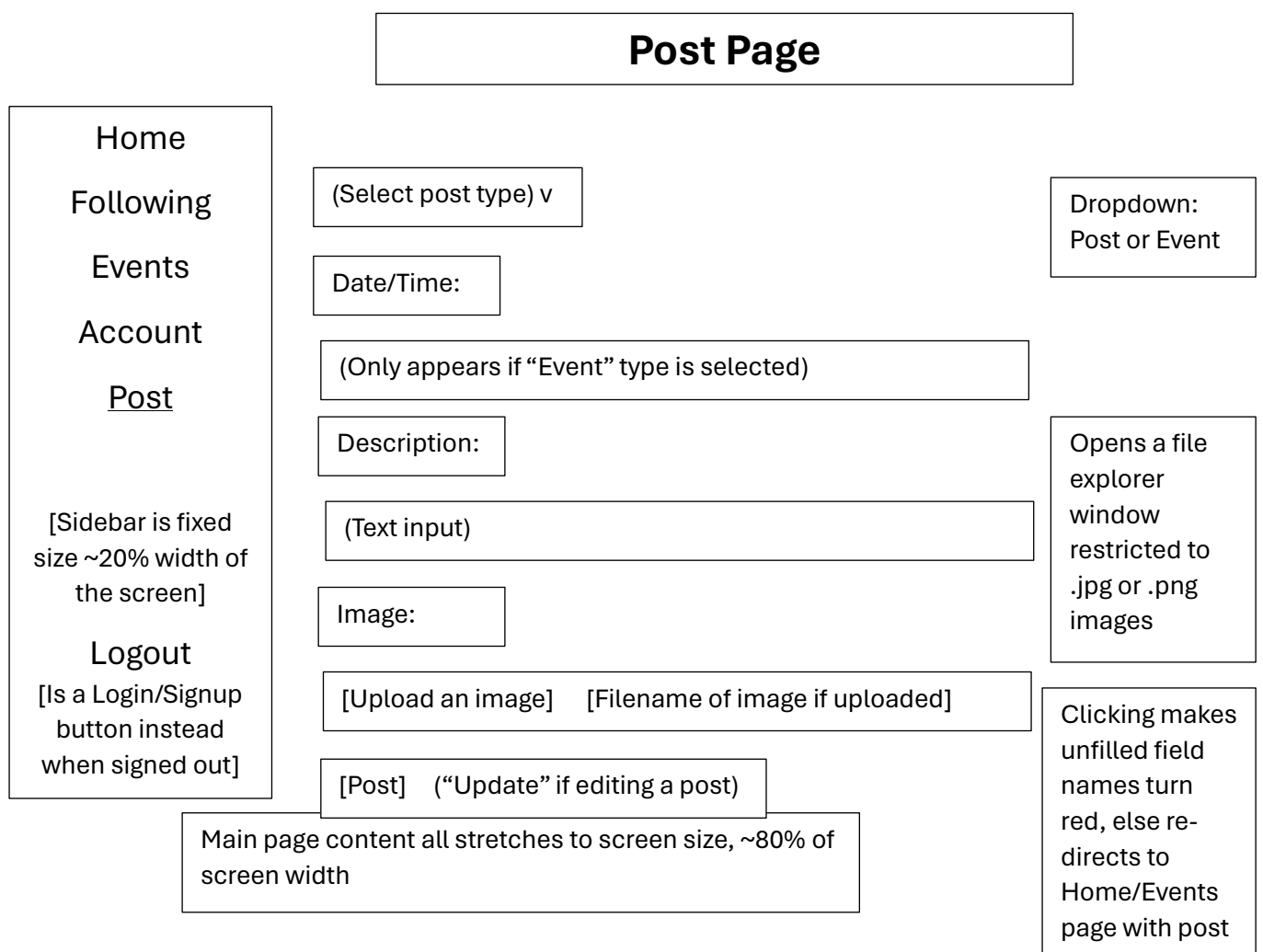
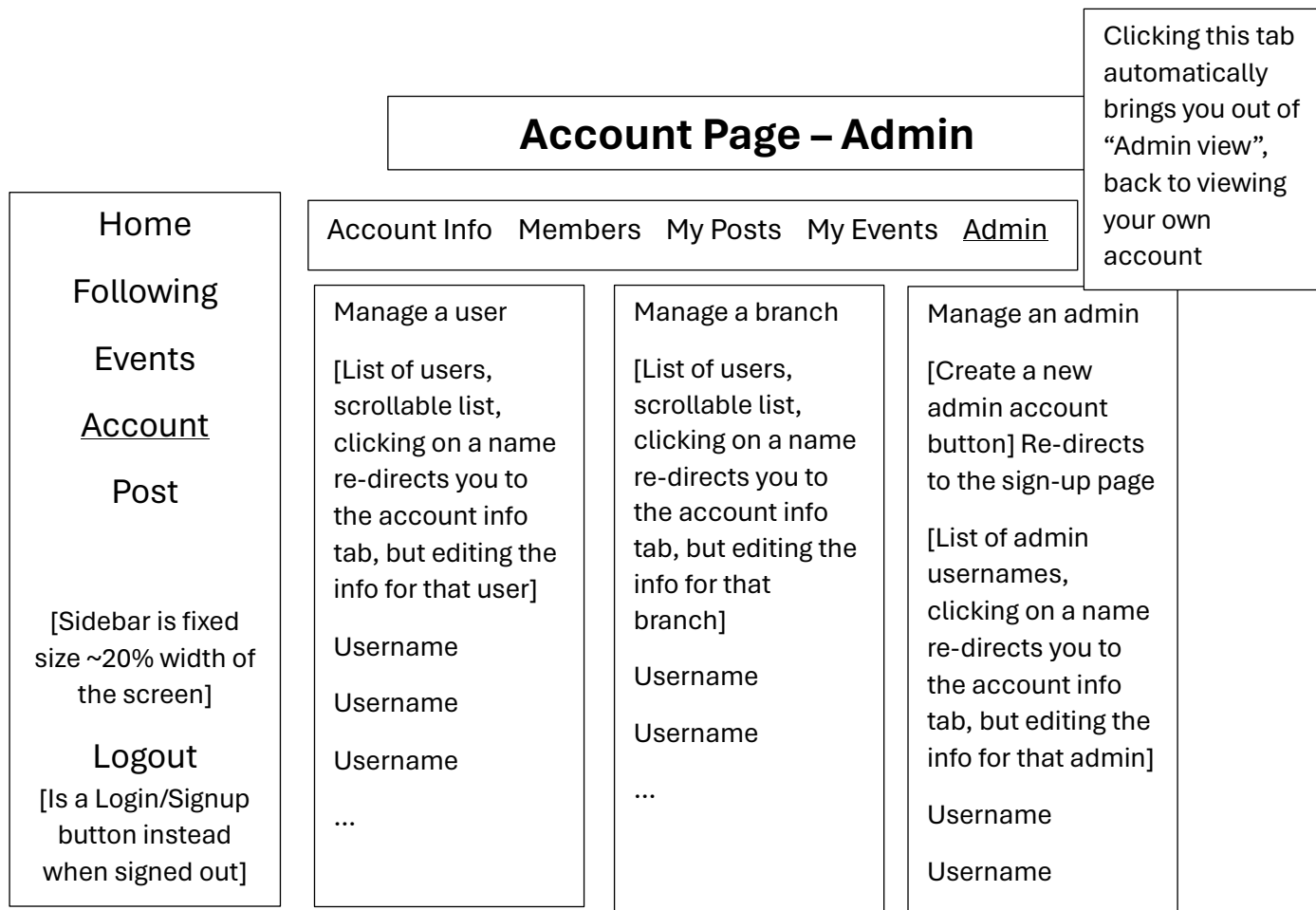
Name

Name

...

[This column/"See
RSVPed" button
does not exist on
the "My Posts" tab]

Main page content all stretches to screen size, ~80% of
screen width



Login Page

Home

Following

Events

Account

Post

[Sidebar is fixed size ~20% width of the screen]

Login/Signup

[Is a Logout button instead when signed in]

Email:

Must be unique

Password:

Hidden text field (*****)

[Login]

Incorrect login details

Login button makes unfilled field names turn red, else redirects to home page

Don't have an account?
Sign up!

Error text appears (in red) if login info incorrect

"Sign up!" link text redirects to sign up page

Main page content all stretches to screen size, ~80% of screen width

Signup Page

Home

Following

Events

Account

Post

[Sidebar is fixed size ~20% width of the screen]

Login/Signup

[Is a Logout button instead when signed in]

Email:

Must be unique

Full name:

Can be anything

Password:

Hidden text field (*****)

Confirm password:

Dropdown:
Individual or Branch

(Choose an account type) v

Signup button makes unfilled field names turn red, else redirects to home page

[Sign up]

Email already used choose another

Error text appears (in red) if email is not unique

Review

The design reduces cognitive load by having a large sidebar that is easy to understand and does not change when navigating through the different webpages. This allows for an intuitive use by the user. Another way cognitive load is minimised is by having a title for each webpage allowing the user to easily identify if the webpage they are on is the one they wanted. As there is consistency through all the webpages it makes it easier for the user to understand and scan the webpage easily producing a webpage detailed on minimising cognitive load. To reduce kinematic load the design has made larger buttons and drop-down menus so that they are easier to be clicked on by the user. As a design we believe the user is likely to want to traverse to different pages often. This is why it was decided to have a sidebar instead of a bar at the top because having buttons on the edge of the screen reduces kinematic load. Further minimising of kinematic and cognitive load comes from the branch managing page where there is a top bar which only allows for the designated tab to be viewed. This allows for the user to reduce kinematic load by clicking on a new tab instead of having to scroll down the page to view all the information wanted.

For the design of the html file, we will be using semantic tags to help to make it more accessible for people with disabilities. According to the WCAG guidelines there are four principles. For perceivable the webpage will make use of images to make it easier for people to see and make sure the colours used do not clash and are easily viewed by people who are colour blind. The webpage will be operable as all links, buttons and menus work and are easy to use with a mouse or keyboard. For understandable the website will be constructed so it is organised and easy to understand for anyone. The language used will be concise and allow people of all ages to read. The website will be robust which means the HTML, CSS and JavaScript will adhere to web standards.

Feedback from another group

The feedback from the other group was that the design was not entirely easy to understand as there was not any dimensions for the page and how the design was going to fit on the page. It was said that the design should be further constructed so that it takes up the whole page to have a clearer understanding of what was designed. Another piece of feedback was that having the login/signup and the account button on the bottom left was not as intuitive to the user as normal convention has them in the top right of the page. The design was also said to not be clear with the colours and stylistic elements of the page as they were not mentioned in the parts of the design. The last piece of advice was that there should be a heading over all the parts of the webpage where the user must input as in the design there is no heading or explanation over what the fields were meant for.

Data Plan

Posting

A post will take a single row of a “Posts” SQL table. Both events and regular posts will be stored in this table, with an enumerator value to distinguish them. 0 = Post, 1 = Event. Using an enumerator data type rather than a bool lets us introduce more post types in the future without having to change the data type of the post type column.

A post description will be limited to 1000 characters.

All post information comes from a user-submitted POST request initiated from the Posts webpage submit button to a special url, e.g. “/posts/submit”. A server route will handle the request rather than there being an actual HTML webpage.

Data	ID	Post Type	Date/Time	Description	Background image path	Author ID (Foreign Key)
SQL Data type	BIGINT	ENUM(0,1)	DATETIME	VARCHAR(1000)	VARCHAR(100)	BIGINT

Posts are retrieved from the server as JSON objects to the client. The client goes to the Home, Following or Events page, the client’s javascript then sends an AJAX request to the server to a specific url e.g. “/home/load_posts”, the server performs an SQL query to find the latest posts, and order them by latest via the Date/Time SQL table column value, the server responds with a JSON object for the client to fill in the webpage with the latest posts.

Post background images

A post background image will be stored as a .png or .jpg file in the public folder file path on the server E.g. public/assets/posts/backgrounds/image.png

All posts are publicly viewable without needing an account, therefore all post background images are public too. As the images are in the public folder, the client simply needs to send a GET request for the image path once they have the post’s data.

User Info

User info is first submitted to the server through a route, sent via a POST AJAX request sent from the Signup webpage from the client, on clicking the “Sign Up” button. A specific url is used for the route/AJAX request e.g. “/signup/create”. The server then handles storing all user info into the SQL server, storing it in the “Branches” table if the user made a branch account, or “Users” if they made a regular user account.

When viewing account info from the account page, to autofill the account info boxes, the client will need to send an AJAX request to the server, “/account/info/get”, and a route will respond with a JSON object after the server performs a SQL query, containing all the info to fill into the page elements. A similar method is used for the “Members” tab of the account page, if accessing this tab as a branch account.

If a user tries to access the Members, My Posts or My Events pages through “/account/info/get”, then the server will respond with a 404 not found.

Administrator Accounts

Admins can make regular users admins by adding as a member of the special “Admin” branch, which acts like a regular branch account. This branch account will be manually created and added to the SQL database on the server as part of the server setup. When accessing the

Account webpage, the client makes an AJAX request to a url like “account/admin_check”, which responds with extra webpage content in the form of HTML or JSON to add the “Admin” button to the top bar of the page.

Trying to send a GET request to the admin tab of the account page will result in a 403 Forbidden response, unless the client is able to send an appropriate session ID to a logged-in admin account as a URL parameter.

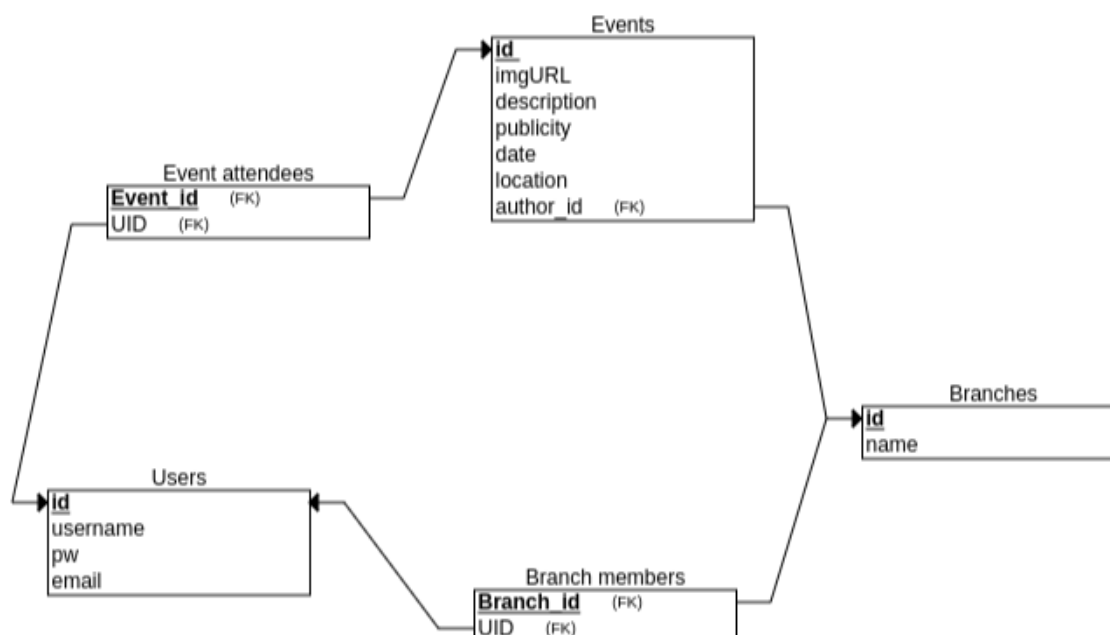
Logging In

After creating an account or logging in from the “Log In” button from the login page, the server should process the login from a URL like “login/process_login” and a route should send back either a JSON response to say the login was not successful, or respond with a re-direct to the Home page, and send a session ID for the client to create a cookie to store on their machine.

Trying to access any of the account page tabs, the Post page or the Following page while not logged in will redirect you to the login page. The server will simply check its active sessions and see if the user is logged into an account with an active session whenever one of these pages are requested. If they are not logged in, the server will respond with a re-direct via a middleware function for all these pages.

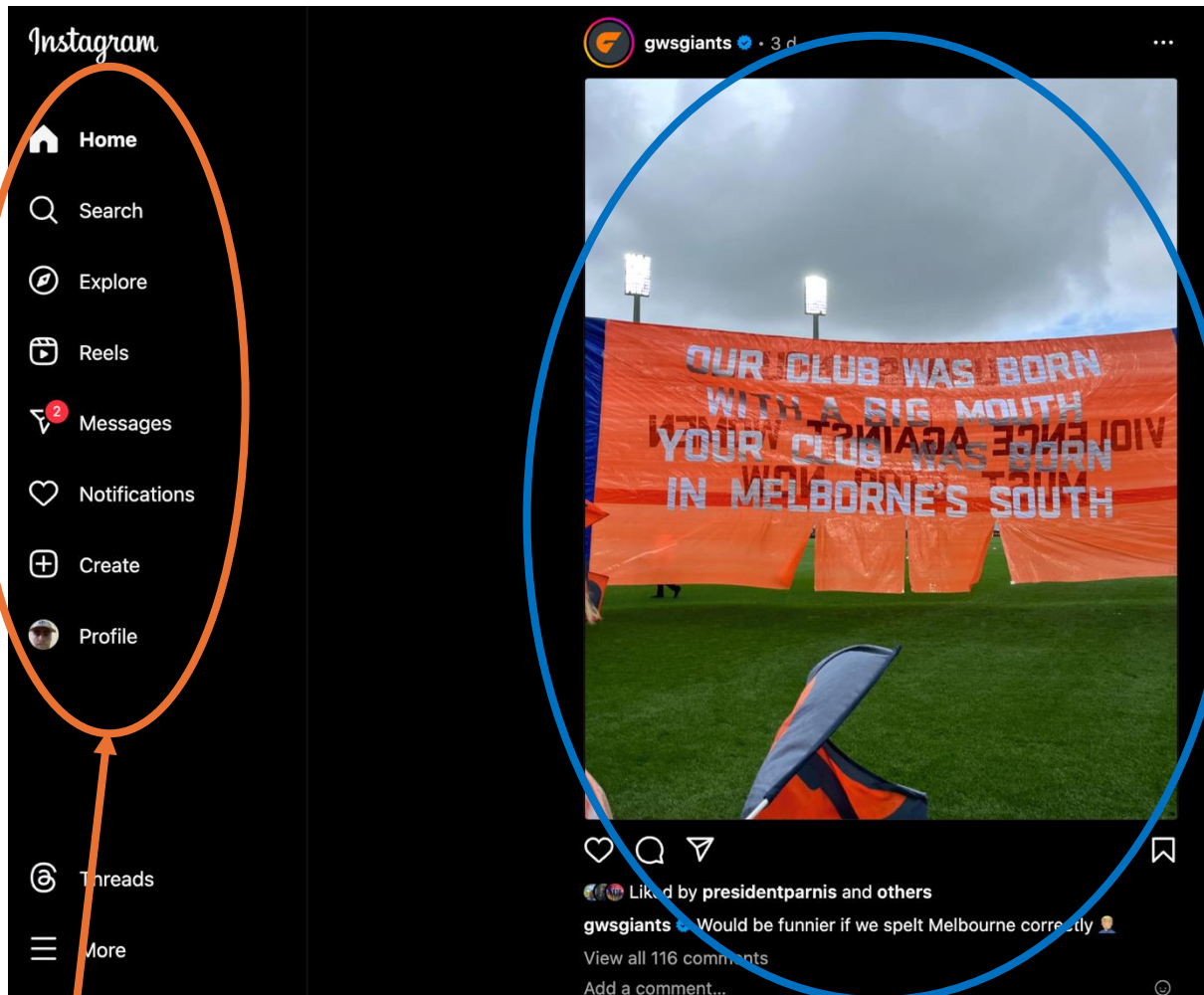
Accessing the Home or Events pages will perform a similar check but will only update the page with “Edit” buttons on posts if the user made that post. The server will send an extra value in the JSON object to indicate the post belongs to the logged in user and the client will display the edit button. A server route “/post/edit” will handle any AJAX requests from the client to edit a post to either respond with forbidden 403, or if they are the user and logged in, respond with a re-direct to the Post page with the post’s current details auto filled.

Database Schema



Research for Possible Website Design/Features

Example 1: Instagram (website display)



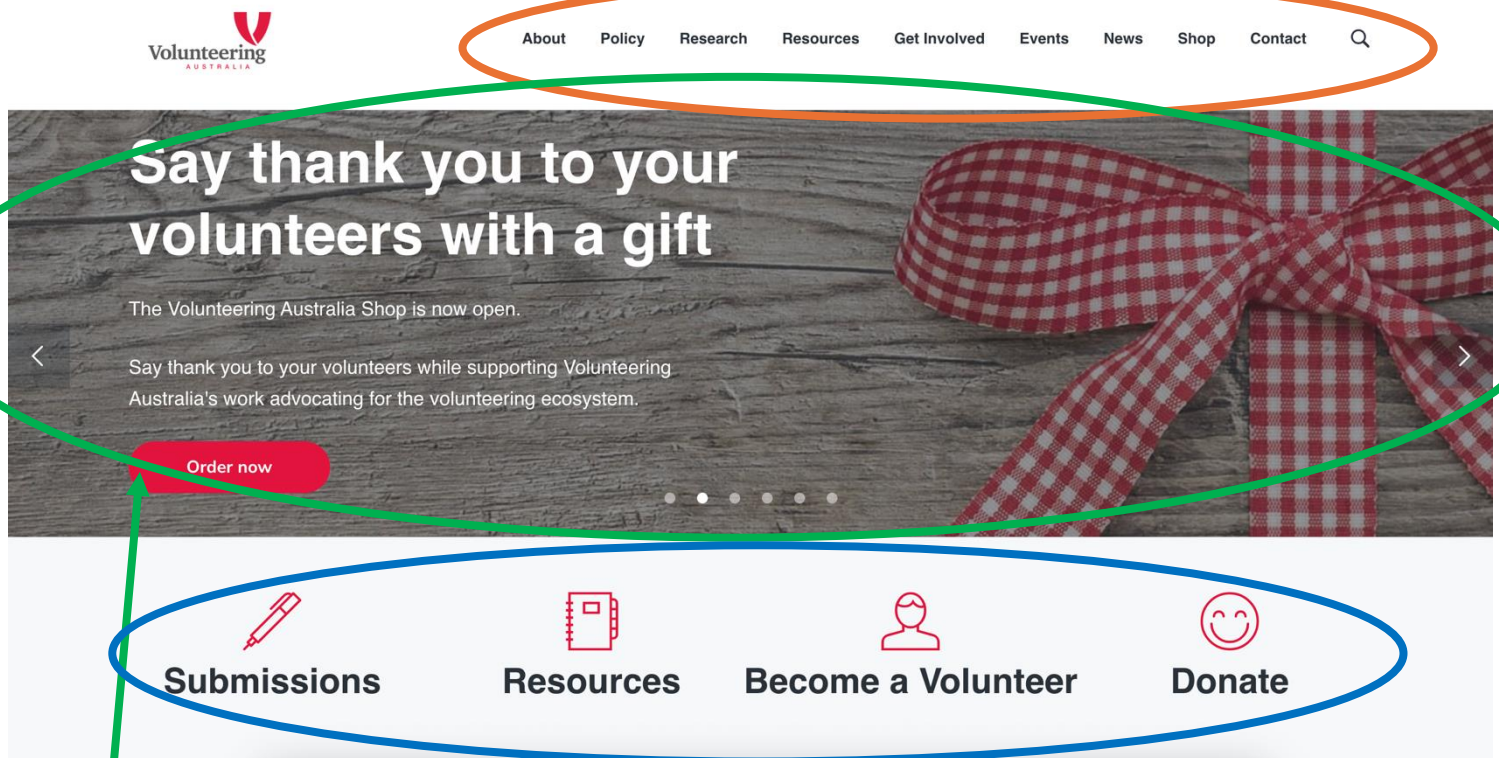
As is highlighted, the different features of Instagram are displayed down the left side of the page. This helps the user find the necessary feature straight away, as they don't need to look hard to find each feature, thereby reducing the associated cognitive and kinematic load.

The blue circle above references the way individual posts/information is shared on Instagram. The posts are purposely made much bigger than other website elements, drawing the attention of the user to each post. The size of each post also reduces the associated cognitive/kinematic load for each user. Furthermore, it can also be seen how each post has a caption, describing the post, and the ability for comments to be added. Both features could be useful for our website application, and thereby are worth considering.

Summary: Instagram is a great base case for us to draw comparison off. The design is clearly considered, such that user cognitive/kinematic load is reduced, making the application incredibly user friendly. Additionally, the features included, such as posting, liking, following and captions are all features which, if used on our website, may increase overall accessibility, increasing the quality of our website as a whole.

Example 2: Volunteering Australia (website display)

As highlighted on this website, the associated features/user options are displayed across the top of the website, in a left to right fashion. Whilst this is still an effective technique, the user may be less inclined to look at the options in the top right corner of the page, as this is one of the last places a user looks on a website, which may be an issue. Despite this, the features are listed in an ordered fashion, and still help provide an insight into what different services this website provides.



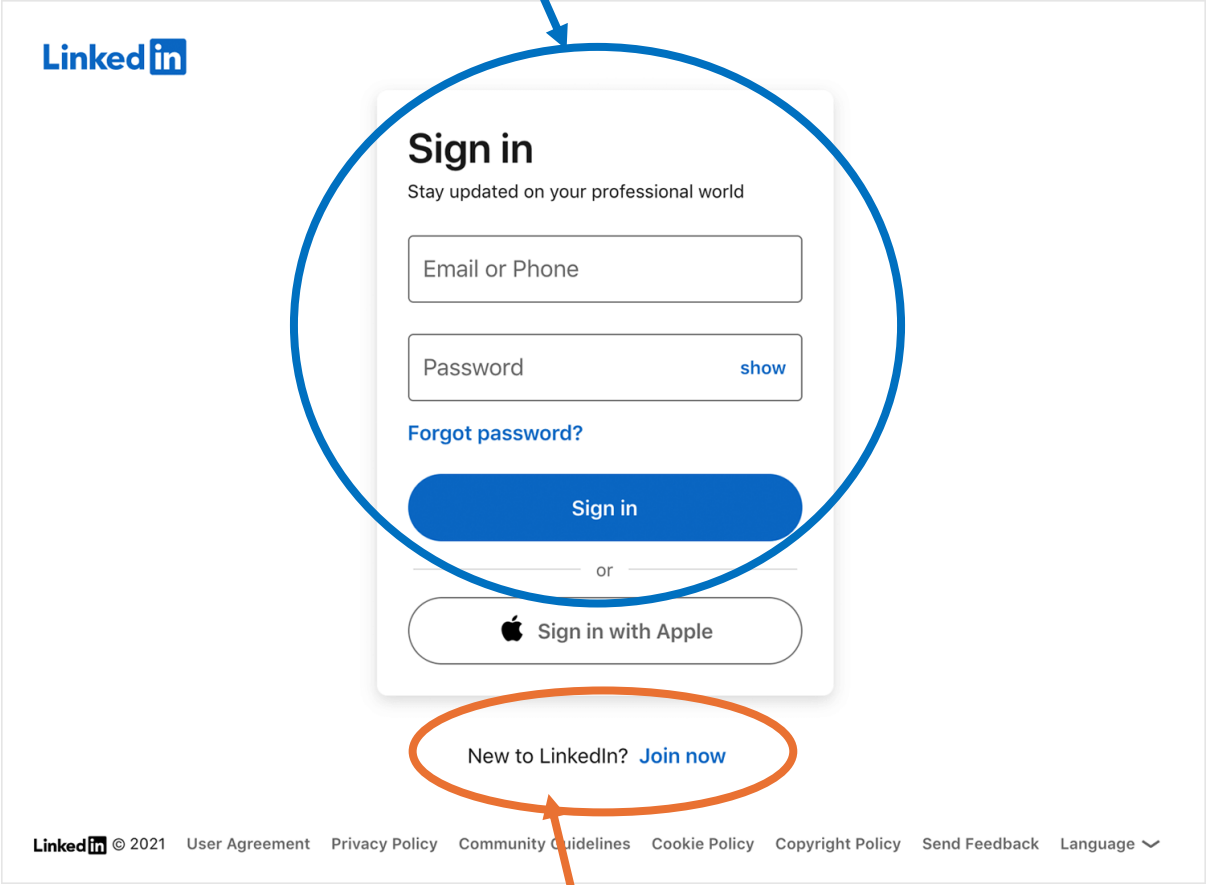
As is circled, this website includes a generic but bold image, right in the centre of the page. This adds a nice aesthetic to the website, as it breaks up the text on the page, and adds some nice colour. This could be a good idea to include on the home page of our website, as the home page should look the same for each user, meaning a general picture could be used.

This website also includes these four graphic elements, each of which, when clicked, take the user to an associated linked page. Through the use of such graphics, the website is able to add another different visual element to the website. Additionally, each graphic also provides the user with a visual representation of what each associated page is related to, thereby increasing overall website usability, making this feature definitely one worth considering.

Summary: The visual design of this website is very appealing. Through the use of visual features, such as images and smaller graphics, the user is made to feel at ease, and is not bombarded with large amounts of text. This website is a great example which we may be able to reference, when designing a possible homepage, which would have the same information for each user, possibly before they login.

Example 3: LinkedIn (login display)

Summary: Highlighted below is a possible login page design. In this example, the login layout is very simple, and thereby very easy to use. The login task is located right in the centre of the page, and each box has a clear label, informing the user of the necessary action to proceed. There is also a notable lack of any other features on this page, making it hard for the user to get distracted/side tracked, reducing the total cognitive load.



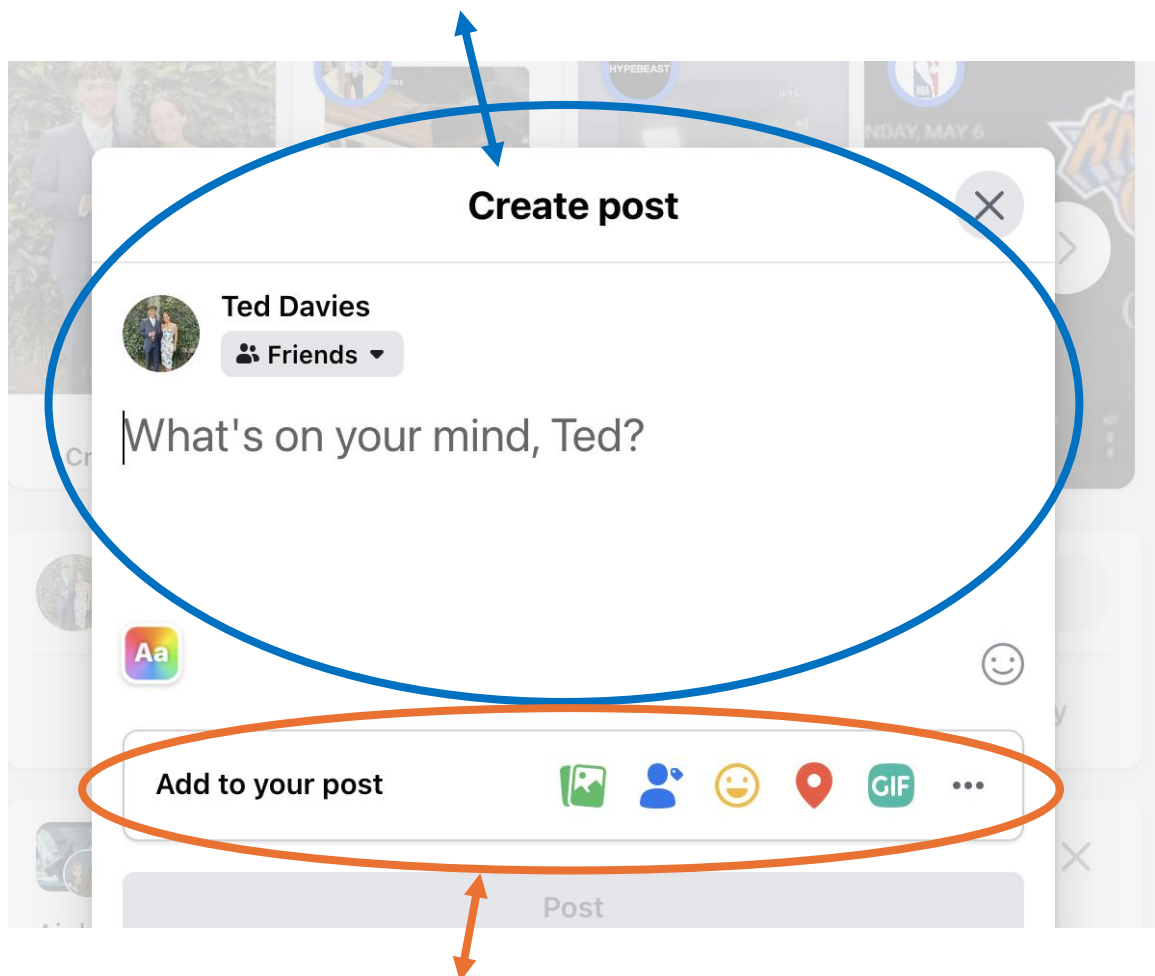
The image shows a screenshot of the LinkedIn login page. A blue circle highlights the main login form, which includes the LinkedIn logo, the 'Sign in' heading, the subtext 'Stay updated on your professional world', input fields for 'Email or Phone' and 'Password' (with a 'show' link), a 'Forgot password?' link, a blue 'Sign in' button, and an 'or' separator. Below the circle, an orange circle highlights the text 'New to LinkedIn? [Join now](#)'. At the bottom, a footer contains various links: 'LinkedIn © 2021', 'User Agreement', 'Privacy Policy', 'Community Guidelines', 'Cookie Policy', 'Copyright Policy', 'Send Feedback', and 'Language' with a dropdown arrow. Two arrows originate from the text boxes: a blue arrow points from the first box to the blue circle, and an orange arrow points from the second box to the orange circle.

Another aspect on this page, which is worth noting, and which may be applicable to our project is what is highlighted in the orange circle. This text provides an option for a user who does not have an account, to then sign up. This is a good feature to include, as it allows user who have not signed up, to sign up, and join our website.

Summary: This webpage is much simpler than the previously analysed webpages, but this simple design does not necessarily mean that it is worse. Through such simple design, the user has a very clear idea as to what the objective of this webpage is. Furthermore, they are able to complete this objective with a high level of ease, as there are no other elements which may distract them. Finally, the use of colour on this page is also worth noting, as the key elements are blue, making them stand out, indicating their importance.

Example 4: Facebook (login display)

Summary: This is an example from Facebook for a possible 'create post' page. Similar to the above example from LinkedIn, it is notably simple, without many different colours, texts or options. As above, this is likely done purposefully, to ensure that the task at hand is as simple for the user as possible, reducing total cognitive load. It is also worth mentioning the provided prompt – 'What's on your mind, User', as this calls the user into action, inviting them to share what is on their mind.



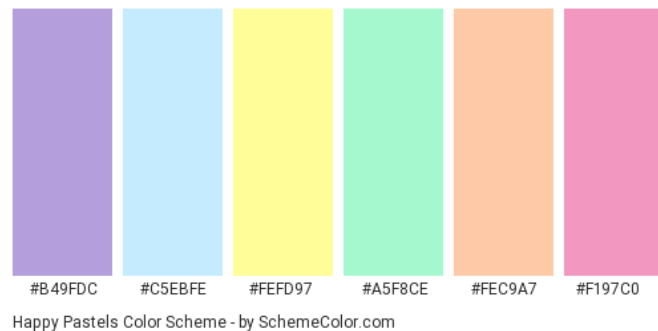
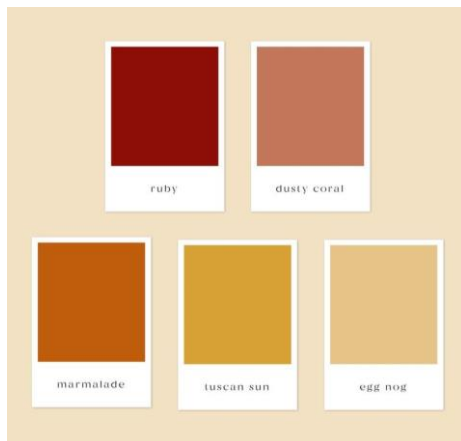
This orange circle references the different options that a user can explore when creating a post, such as adding an image, emoticon or GIF. This is an important element to this page, as it is a link to all the different features associated with a post. Whilst these features are included here, they are displayed in a very small manner, without any captions, which does increase the cognitive and kinematic load for the user, as it may not be initially clear what each of these options do and how to find them.

Summary: Similar to LinkedIn, this webpage example is quite simple, as the user task is not overly complicated. Despite this, this may be an example where the design/layout is too simple, as it is not explicitly clear for the user as to how to incorporate different features into their post, making that a possible room for improvement.

Style Aspects:

Colour Pallet:

For our website, as it is a volunteer-based website, it is important that the colours chosen to concur with the overall theme of the created website. Thus, the colours should be warm and welcoming for the user, inviting them to stay on the website and explore different volunteer branches. The colours should not be overly harsh, and should blend in well with one another. Some possible examples are shown below:



Fonts:

For similar reasons to those shared above, we want the fonts used on this website to be gentle, easy to read and soft, thereby adding to the overall theme of the page. Some possible font options are shown below:



Feature Outline and Discussion

Within our website, there are multiple different features, each of which ensure that the website provides the user with the best possible experience, whilst simultaneously allowing the website to operate with a high level of functionality. Some of these features are listed and described below:

SIGN UP OPTION:

- Whilst signing up, the user will be able to put in their associated details, which will then be stored in the system.
- The user will also be able to choose their account type, regarding possibly being a branch manager, or just a standard user, which will thereby then dictate the features which they can access whilst using the web service.
- If a certain option isn't filled during the sign-up process, an error will be displayed, forcing the user to make an entry in each field.

Additional to the above, there will also be a login page similar to the sign-up page, where, once a user has created an account, they can then use this page to login in straight away. It will require a username and password.

POST FEATURE:

- For an account which has a certain level priority associated with it, they will be able to make a post. These posts will be volunteer events, which are shared to the users to inform them of up-and-coming events which they can attend.
- Each post can have an associated image, text, date/time and location.
- Each post should be able to be seen by users who are following this branch. These users should also be able to like and RSVP to such posts.

ADMIN CONTROLS:

- As stated in the task description, administrative accounts should be able to manage users and their accounts, manage branches, and approve new admin joining the website. The admins on our website will be able to do each of these tasks.

USER ACCESS FEATURES

- The user will be able to RSVP to volunteer branch events. After the user has RSVP'd to multiple events, the user will be able to access a page, where their associated events are all listed, informing them of their upcoming commitments.
- Furthermore, each user should be able to follow/join various volunteer branches. Through doing this, they will then be able to see the new posts from this branch, and like and interact with these posts.
- The user may also possibly be able to view 2 types of feeds, one which they only see posts from the branches which they are following, and another where they see posts from all different branches.