Contents

[Use Case Diagram 2](#_Toc157427510)

[Hierarchy Diagram 3](#_Toc157427511)

[Algorithms 5](#_Toc157427512)

[Register Algorithm 5](#_Toc157427513)

[Login Algorithm 5](#_Toc157427514)

[Air Pollution Algorithm 6](#_Toc157427515)

[Weather Algorithm 7](#_Toc157427516)

[Advice Generation Algorithm 8](#_Toc157427517)

[Wireframes 8](#_Toc157427518)

[User Pages 8](#_Toc157427519)

[User Features Page 9](#_Toc157427520)

[Weather Forecast Page 10](#_Toc157427521)

[Air Quality Dashboard Page 11](#_Toc157427522)

[All Advice Page 11](#_Toc157427523)

[Saved Advice Page 12](#_Toc157427524)

[Advice Details Page 13](#_Toc157427525)

[Management Pages 13](#_Toc157427526)

[Management Features Page 14](#_Toc157427527)

[Manage Advice Page 14](#_Toc157427528)

[Delete Advice Pages 15](#_Toc157427529)

[Universal Pages 16](#_Toc157427530)

[Welcome Page 16](#_Toc157427531)

[Register Page 17](#_Toc157427532)

[Log In Page 17](#_Toc157427533)

[Report Bug Page 18](#_Toc157427534)

# Use Case Diagram

The use case diagram is intended to illustrate each of the roles in the web application, what features they have access to and how each of the features are hosted. In this solution, there are only 2 roles planned – user and management. However, should Health Advice Group require more roles and stricter permissions, we will modify the permissions and access to features as needed.

A diagram of a company

Description automatically generated

# Hierarchy Diagram

The hierarchy diagram below is designed to show the flow of the website. User and management will have access to different features or different parts of a system, which is why the hierarchy diagram splits into 2 different paths. As of right now, users have read only access to the advice system, weather forecasting and air quality systems. However, management will have read and write access to the advice system and will have access to a dashboard to measure user interactivity with advice and the solution in general.

A diagram of a company structure

Description automatically generated

# Algorithms

## Register Algorithm

To register a user, the data entered must be accurate as per data protection regulations. This means that data entered must undergo rigorous validation. Attached below is the decision-making process for whether a user signing up is valid, or if details are invalid or need to be re-entered.

A diagram of a flowchart

Description automatically generated

START

WHILE TRUE

GET first\_name FROM KEYBOARD

IF first\_name > 2 AND first\_name < 20 THEN

BREAK

WHILE TRUE

GET last\_name FROM KEYBOARD

IF last\_name > 2 AND last\_name < 20 THEN

BREAK

WHILE TRUE

GET email\_address FROM KEYBOARD

IF email\_address CONTAINS “@” THEN

IF email\_address CONTAINS “.” THEN

BREAK

WHILE TRUE

GET password FROM KEYBOARD

IF LENGTH(password) >= 8 THEN

IF password CONTAINS characters.lowercase AND password CONTAINS characters.uppercase AND password CONTAINS characters.numbers AND password CONTAINS characters.symbols THEN

BREAK

WHILE TRUE

IF confirm\_password EQUALS password THEN

BREAK

END

## Login Algorithm

In order to protect against security risks such as brute force attacks, the login system must still check against multiple values and conditions before a user can access their account and the platform. Below is the decision-making process behind whether a user will be granted access to the system.

A diagram of a flowchart

Description automatically generated

START

WHILE TRUE

GET email\_address FROM KEYBOARD

GET password FROM KEYBOARD

HASH password WITH SHA256

GET FROM UsersDB

IF lockout\_enabled IS NOT TRUE

IF email\_address IN DB

IF hashed\_password IN DB

BREAK

END

## Air Pollution Algorithm

This indicates how we will interact with an air pollution API to input data and output results for display to the user. It relies on the user to enable their location so we can send their latitude and longitude to the API.

A diagram of a flowchart

Description automatically generated

START

GET latitude FROM USER LOCATION

GET longitude FROM USER LOCATION

CONNECT TO OPEN WEATHER API

SET API DATA TO latitude, longitude

DISPLAY pollutants TO USER

FOR EACH value IN POLLUTANTS DO

PLOT value TO GRAPH

COMPARE value TO pollutant\_range

DISPLAY COMPARISON FROM pollutant\_range TO USER

DISPLAY GRAPH TO USER

END

## Weather Algorithm

This indicates how we will interact with a current weather API to input data and output results for display to the user. It relies on the user to enable their location so we can send their latitude and longitude to the API.

A diagram of a process

Description automatically generated

START

GET latitude FROM USER LOCATION

GET longitude FROM USER LOCATION

CONNECT TO OPEN WEATHER API

SET API DATA TO latitude, longitude

GET FROM AdviceDB

DISPLAY current\_temperature, location, condition, sunrise, sunset TO USER

FOR EACH advice IN AdviceDB DO

IF advice.temperature – current\_temperature < 5 AND IF advice.temperature – current\_temperature > -5

DISPLAY advice FROM TO USER

END

## Advice Generation Algorithm

Whilst there are multiple variations of advice being generated, for instance when the temperature matches that of a condition, when a user is searching for advice which matches a title, or when a user is accessing saved advice, this is the simplest version which will unconditionally fetch all advice.

A diagram of a flowchart

Description automatically generated

START

GET FROM AdviceDB

IF AdviceDB IS NULL DO

SEND “NO CURRENT ADVICE” TO DISPLAY

ELSE DO

FOR EACH advice IN advice\_db DO

generate\_article(advice)

END

# Wireframes

In order to assist in the development process, I have added design documents to showcase the layout of the website and roughly where each option will be.

## User Pages

These are the pages which the user will be able to see should they be logged in and assigned the “user” role.

### User Features Page

This is the page which the user will see initially after logging into the system, and it is intended to list all the features available to them.

A screenshot of a computer

Description automatically generated

### Weather Forecast Page

This is the page the user will see after clicking the “weather” option. It will by default use the user’s current location, but it can also work with an entered location.

A screenshot of a computer

Description automatically generatedSearch Another Location Page

This is the form the user will need to fill out if they want to access the weather at another location. This will be translated into a latitude and longitude by the API.

A screenshot of a computer

Description automatically generated

### Air Quality Dashboard Page

This is the page the user will see after clicking the “weather” option. It will by default use the user’s current location, but it can also work with an entered location.

A screenshot of a computer

Description automatically generated

### All Advice Page

This is the page the user will see after clicking the “advice” option. It will show the most recently added advice first by default.

A screenshot of a computer

Description automatically generated

### Saved Advice Page

This will show the advice that a user has saved, for quick and easy navigation.

A screenshot of a computer

Description automatically generatedHealth Tracker Page

This is the page the user will see after clicking the “health tracker” option. It is designed for adding steps, calories, or water as the day progresses and keeps a cumulative total.

A screenshot of a fitness tracker

Description automatically generated

Advice Details Page

By default, the details / description of advice will not be in on the page to keep the initial advice page clean. Should a user want to see more details, they can click the details button and it will lead them to a page which looks somewhat like this:A screenshot of a video player

Description automatically generated

## Management Pages

These are the pages which the trustees will be able to see should they be logged in and assigned the “management” role.

### Management Features Page

This is the page which the user will see initially after logging into the system, and it is intended to list all the features available to them.

### A screenshot of a computer Description automatically generatedManage Advice Page

This is what a management user would see upon clicking the “manage advice” page. It has a similar view to the user, with the addition of the edit and delete buttons for each advice article.

A screenshot of a computer

Description automatically generated

### Add / Edit Advice Page

Editing and adding advice will use roughly the same form and have a similar layout.A screenshot of a computer

Description automatically generated

### Delete Advice Pages

A screenshot of a computer

Description automatically generated

## Universal Pages

These are pages that both “user” and “management” roles share or pages which can be accessed before signing in.

### Welcome Page

For data protection, monitoring, and performance purposes, users cannot access any features until they have registered and / or signed in. This is roughly what an unauthenticated user will see upon opening the site.

A screenshot of a computer

Description automatically generated

### Register Page

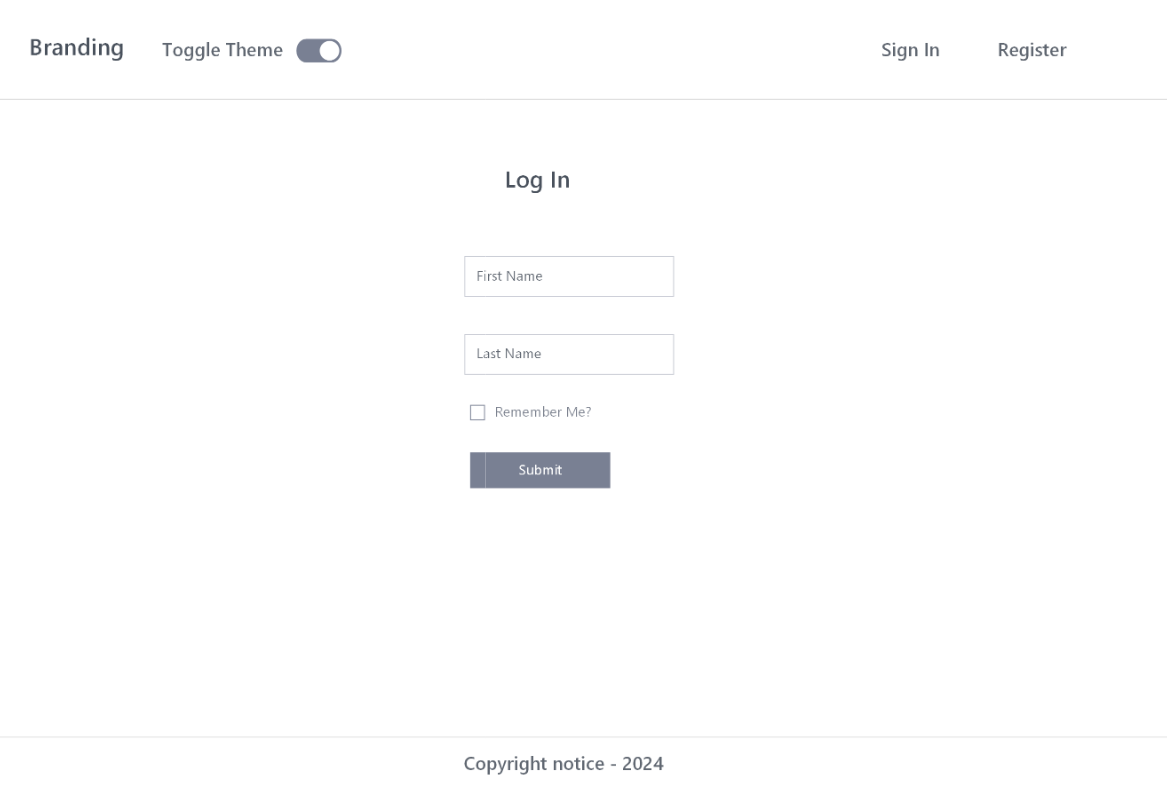
When a user wants to create an account for the service, this is the data they will have to enter.

A screenshot of a login form

Description automatically generated

### Log In Page

Once a user has registered, these are the details which they will need to enter to access their account and successfully be authenticated to access the service.



Report Bug Page

Should a user need to report a fault with the system, these are the details which they would need to enter. A screenshot of a login form

Description automatically generated