Health Advice Group

# Project Proposal

# Contents

## Overview

Health Advice Group have requested a software that can provide accurate, thorough and personalised information on weather health effects as a result of weather conditions. This will include the potential for allergies, personalised warnings for allergen risks and advice on how to cope with extreme weather conditions.

The project will be carried out over a period of 4 weeks with 30 office hours provided for development. The project will begin with the front-end development that will be modelled on the front-end design documents provided. Front-end is planned to be completed in 15 hours over the first 2 weeks of development. The front-end development is expected to be the bulk of development as it will provide all the static data such as information articles required within the software.

Upon completion of the front-end development, back-end development will begin. A database that will be used to store data on the required personal health tracker will be developed along with the API required to access weather and pollen data for the front-end. Back-end development is expected to take 10 hours over 1 and half weeks to complete.

The final section of development to be focused on is accessibility features. This will include ensuring all aspects of the site are readable both by humans with ease and screen readers for people unable to view content effectively. The expected time frame for accessibility development is 5 hours over half a week.

The development of a weather and pollen health tracker will be of benefit to customers that suffer from weather or allergy related ailments such as hay-fever. Providing support and advice that will help customers make informed decisions on strategies they can take to avoid being encumbered by their ailments.

# 1.0 Business Context

## 1.1 Stakeholders

| Stakeholder | Relation | Desired Outcome |
|---|---|---|
| Customer | The customer will be the primary beneficiaries of the project. The customer will take advantage of the services provided. | The customer will expect a functioning service that they can use to access accurate weather data and advice on allergy risks for a given location and time. They will also expect information on how to cope with extreme weather conditions. |
| Developers | The developers of the project will execute the project. They will be behind the creation of the project ensuring all user requirements and client requirements are met. They will also be behind the maintenance and further development of the final product in future as the software continues to exist and expand. | The developers will be employees of the project therefore they will expect financial compensation for the development of the project. |
| Health Advice Group | Health Advice Group is the client and the sole investor in the project. They will be the public facing representatives of the project and have their brand labelled as the primary provider of the final product. | Health Advice Group will expect a high-quality product that they can market to provide positive attention to their company and more traffic to their services whilst maintaining cost efficiency. |

## Customer

Customers will be the primary users of the final product. To ensure their continued custom they will expect a product that can provide a benefit to their way of life and be easy to use. Customer satisfaction will be the primary focus of the investors. The investors of the project will not see any returns if the customers are not satisfied and not using the product. Therefore, the needs of the customers and the accessibility of the product to as many customers as possible is important to ensure that not only the customer receive their desired outcome but the investors as well.

The power of the customer is high after the project has been deployed and their feedback becomes the key indicator of the success of the project. However, during the initial development of the project the customer will have minimal power over the project. The interest in the project by customers will be high throughout as they will be the primary beneficiaries of the final product.

## Developers

The developers will be executing the project and creating the final product. They will be expected to focus on the clients' requirements and ensure that the client is satisfied with the final product. Developers will have the lowest interest out of all stakeholders in the final product of the project however, they will have the highest involvement in the project. Developers desired outcome will be financial compensation by the client for the development of the project. Developer's power over the final product will be minimal as they must conform to the clients' requirements. Developers will be able to make informed suggestions to the clients and discuss the feasibility of client requirements however, they will not have the final say on the outcome of the product.

Possible negative outcomes for the developers throughout the project may come from any delays within the development process. This could cause a delay on the due date for the project and therefore, a delay on the payment for the project. The developers' reputations will also be at risk if the project does not develop smoothly potentially hindering future endeavours for the developers.

A positive outcome for the developers will be a successful project development phase that continues smoothly without hitting delays and successfully meeting the deadline.

## Health Advice Group

Health Advice Group is the client for the project and the sole investor in the project. Health Advice Group will lead the requirements for the project and monitor how the expectations of the project are being met. Health Advice Group will expect a high-quality product that meets all their requirements and can be marketed for the companies benefit. Health Advice Group will want a product that users enjoy as they will be receiving the benefits of having users that want to use their product.

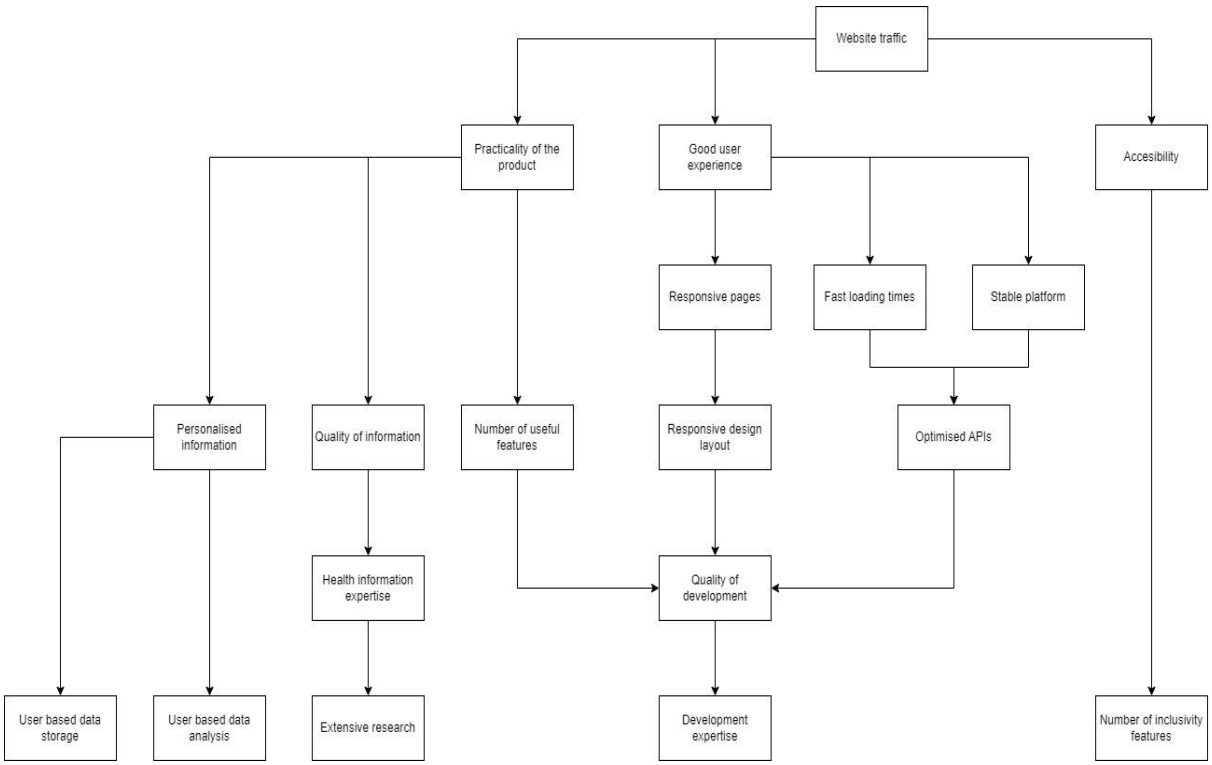
Health Advice Group will have the highest power over the outcome of the project. They will have the final say on all feature implementations and designs for the project. Health Advice Group will pre-approve all aspects of the project before the final product can be launched. As well as having the highest power of the project they will also have a high interest in the outcome of the project. Therefore, Health Advice Group are the most important stakeholder to ensure their satisfaction and that their desired outcome is met.

Potential risks for Health Advice Group are reputation loss if the final product is not of high quality and does not represent the company appropriately. Health Advice Group will be the public facing representative of the final product therefore anything associated with the product will have a direct impact on their public image. Another risk for Health Advice Group as the sole investor is loss on return. Failure to market the product or failure to provide a satisfactory service will result in losses for the company as they cannot make a return on their investment.

Positive outcomes for Health Advice Group will be profit on return of the product. A successful product that is adopted by customers well will provide a return on investment for the company. A successful product will also provide a positive impact on the company’s reputation increasing their potential for future business endeavours being successful.

## 1.2 Drivers

### Drivers Relationship Map




The primary goal of the final product is to maximise traffic to the website. The diagram above is a relationship diagram outlining the hierarchy of the business drivers involved in ensuring maximised website traffic.

The practicality of the product, the user experience and the accessibility are identified as being important to the usability of the site, user satisfaction and the likelihood of a user finding purpose in using the site. The key drivers of the product will be user-based storage, user-based data analysis, extensive research, development expertise and the number of inclusivity features.

User-based storage and user-based data analysis will be key to increasing the practicality of the site as it will allow users to make their own informed decisions about their health based on data collected by the site. It will also allow the site to make relevant information more readily available to the user in turn increasing user satisfaction and the likelihood of a user recommending the website.

Extensive research will be necessary to ensure that all information provided by the website is of concise, relevant and accurate. Failure to provide fully backed information could result in mistrust for the website by users and decrease the likelihood of users returning to the website.

Development expertise will be required to ensure optimised website speeds, the quality and number of usable features and responsive designs. Fast intuitively designed websites are key to providing a good user experience necessary to satisfy customers and increase the likelihood of recommendation of the website. Responsive designs allow access to the website from any platform increasing the availability of the website from anywhere and making the experience of using the website seamless from platform to platform. Optimised APIs and statically generated pages where applicable will increase the speed of the site significantly avoiding frustrating users attempting to access the websites services.

The number of inclusivity features is very important in ensuring that all users regardless of physical abilities can access the website with ease and maintain a high-quality experience.

## 1.3 Constraints

### Internal Constraints Table:

| Constraint | Impact of Constraint | Management of Constraint |
|---|---|---|
| Time | The time available for the project is low. 30 hours of office hours provides a tight time frame to develop the product within. This will impact quality as the ability to develop extra quality of life of features will be restricted. | A structured timeline for the project development will be followed to maximise time spent on core software requirements and ensure that client and user requirements are met to their full extent. Extras and optional features will be focused on last if time is leftover. |
| Cost | The budget for the project is £0. Therefore, all features developed, and the services used for development must be free. This will restrict the number of available resources to use when developing the software. | Free services will be used, and all available services will be used to their maximum capability within the context to ensure that the product is as feature rich and robust as possible. |
| Scope | The scope of the project is impacted by the cost and time available. The project is restricted by the availability of resources and the time frame provided to implement these resources. Therefore, the project will have to prioritise only developing key features and avoiding broadening the scope until all core aspects are complete. | The project will solely focus on core features of the client and user requirements until all have been met and made stable. The scope will only broaden to include further development and optional features when core components have been developed. |

| | | |
|---|---|---|
| Quality | The quality of the project will be restricted heavily by the funding and time frame provided. User experience and extra features will have to be put at the bottom of the priority list until the required features are implemented securely. | All core features will be developed to a high quality and with maximised optimisation in order to provide the best user experience possible without implementing unnecessary features. If time and cost constraints allow later into the project, then user experience will be further developed and provided as an optional extra. |
| Resources | The access to resources for the project is limited greatly by the lack of budget. Resources must all be free to access limiting the number of high-quality resources available to the project. Restrictions on services may be required to conform to free access to resources. | All resources available will be maximised and workarounds will be implemented for limitations such as caching data to avoid unnecessary API calls. |

## External Constraints Table:

| Constraint | Impact of Constraint | Management of Constraint |
|---|---|---|
| Legislative | Data laws will determine the software's ability to collect user-based information. The software will only be available as a personalised tracker within the UK and EU as personal data will be stored within the UK and follow GDPR laws. The software will have to provide consent forms to determine whether a user consents to their data being used for the software's purposes before registration to the service. | The ability to register for personal tracking will be restricted to residents within the EU and the UK. |
| Localisation | The software will only be available to use in UK English. This will hinder the accessibility of the software internationally and decrease the number of customers available to use the software. This is as a result of the time and budget constraints not allowing for the service to be expanded to allow for multiple languages support. | The software will have to be done in UK English google translate can be used on sites through google however, the ability to manage this constraint is limited severely by the projects time, resources and budget. |

| | | |
|---|---|---|
| Security | The project will have to account for securely storing people's sensitive data. Personal data will be required to provide personalised services within the software. Therefore, authentication and encryption must be a part of the development phase to ensure safe storage of data. | Authentication and encryption of sensitive data will be used to ensure that all sensitive data is securely stored without risk of personal data breaches. |
| Risks | Storing data on servers will run the risk of server hardware being damaged by unforeseen issues. This could lead to data loss, data breach and in turn customer loss and reputation loss. | Risks can be managed by backing up data to separate servers so that in the event of a server failure data loss is not significant. Data breach risks will be managed through security constrain mitigation via encryption and authentication. |

## 1.4 User Environment

Users will access the final products of the project via a web browser. Web browser support must be numerous for the final product to allow users within different browser contexts to access the services without issue. Restrictions may occur on older browsers as support for features required within a modern web context may not be available.

The user will require a network connection to access the final products services. Accessing the final product will be available through any device that can connect to the internet and has a web browser. Designs will be responsive to allow for a good user experience regardless of device screen size. The software will be hosted on a server accessible via a web browser by the user.

## 1.5 Operational Environment

The final product will be hosted on a third-party server. Server availability will not be within the control of the development team or client. The server will be based within the UK and EU area. GDPR laws will need to be followed in order to ensure the legality of the storage of the data collected by the product.

Third party servers can pose the risk of issues occurring within the hosting companies' systems potentially causing the products services to be unavailable temporarily. However, third party hosting companies often provide many servers to host from. Therefore, if an issue occurs on one server such as a hardware malfunction or it goes down the services should be running again soon after on a different server.

## 1.6 Deployment

The final product will be deployed to a server within the EU and UK. The servers will be provided by a third-party hosting company and deployment will be carried out by the DevOps team within the development team. Testing will be carried out in the deployed environment by the DevOps team and the product will be made live by the end of the 4 weeks dead-line deployment will follow a 4-phase plan.

Initially the product will be developed in a development environment and basic testing will occur as each section of the code base is developed.

Upon development completion a thorough test will be completed following a test plan for the code base ensuring that all features work as expected. After testing has been completed environment variables will be determined and the system will be automated to handle errors that may occur as effectively as possible.

Once the code base has been thoroughly tested, environment variables have been introduced and automation has been completed the code base will be deployed to the live environment where it will be made publicly available and monitored against the KPI’s.

# 2.0 Functional and Non-Functional Requirements

## 2.1 Functional Requirements Table

| ID | Requirement | Mandatory (Y/N) |
|---|---|---|
| F1 | Information explaining how to handle extreme weather conditions. | Y |
| F2 | Information on environmental health conditions | Y |
| F3 | Information on the potential for issues to arise as a result of health conditions at a given time and location. | Y |
| F4 | Email and password-based user registration. | N |
| F5 | User-based personalised health advice based on a risk assessment form. | Y |
| F6 | User-based personalised health advice on provided health conditions for a given location and time. | N |
| F7 | User-based tracking of trends with weather conditions that are relevant to a user's inputted health conditions. | N |
| F8 | A weather forecast widget displayed on an overview dashboard. | Y |
| F9 | An air quality widget displayed on an overview dashboard. | Y |
| F10 | A link to relevant information based on weather conditions viewable on a weather forecast widget on an overview dashboard. | N |
| F11 | Screen reader accessibility. | N |

## 2.2 Decomposition of Functional Requirements

### F1

Information explaining how to cope with extreme weather conditions will be available as articles in the system. These articles will be accessible through both a page displaying all articles in a list linked in the navigation bar and via a link available in the weather forecast widget that links to a relevant article for the current weather conditions.

### F2

Information on health conditions will be available as articles in the system. These articles will be accessible through both a page displaying all health condition information articles in a list linked in the navigation bar and via a link available in the air quality widget and weather forecast widgets that links to a relevant article for the current air quality conditions or weather conditions.

### F3

Information on health conditions based on current time and location will be available as a widget on the main page dashboard displaying the relevant article's summary and acting as a link to the full article. Location data will be based on the user's current location if permission is granted. User location data can be accessed via data provided by the browser and accessed programmatically in JavaScript using the Geolocation API. i.e. navigator.geolocation.getCurrentPosition(). Alternatively, if a user does not provide permission for the application to use their location data. The application will provide a place for a user to enter an address and an external Reverse Geocoding API will be called to turn their address into coordinates that can then be used to access weather data via an external Weather API. The time for the user will be based on the user's current time zone identified through the currently inputted location. The current time zone will be determined through a Time zone API using the latitude and longitude returned by either the Geolocation API or Reverse Geocoding API.

### F4

Email and password registration will be completed via either manual email registration which is stored in a database or through OAuth. Emails will be stored in plain text and passwords will be stored as hashed passwords that are matched against using an encryption key when a user attempts a login. OAuth will be provided by google to login and handles primarily by google with the necessary details stored in the same user database as manual registration to avoid users logging in with OAuth and manual email registration with the same email. Emails will be a unique key and multiple accounts with the same email will not be able to be created by a user. User information will be based on a unique user UUID that is used to determine their access to data within the application.

## F5

A risk assessment form will be able to be completed by logged in users. The risk assessment form will ask general health questions and provide the option to input a health condition that a user is affected by that can be affected by their environment. A pre-set array of health conditions will be provided. This will cause a limitation on the ability to provide data on all health conditions. However, a work around will be a field to provide a custom health condition. Upon entering this health condition, it will be sent to a database to be added as an option by an admin later. Upon completing the risk assessment form the application will provide articles related to the health condition involved and provide notifications to the user within the app when their health condition is of concern on a current date in their current location.

## F6

User personalised health information will be paired between a risk assessment form and the user's location and time information. User location information will work the same for both guest users and registered users. Location will be identified through the Geolocation API on a browser. The Geolocation API will only provide user location data if approved by the user so in the instance a user does not provide their location, they can input an address to use in a text input bar, and a Reverse Geocoding API will be used to determine the address coordinates. The time zone of a user will be determined through a Time Zone API using the latitude and longitude returned by either the Geolocation API or the Reverse Geocoding API. For registered users a default address can be inputted to provide data on a specific location without the need to input the address multiple times or allow the application to access the user location. The default address registered to a user account will be the one used as the default location whenever a registered user accesses the web application.

## F7

Previous weather and environment data for a user will be provided on a line graph for users in an expanded trends widget on the dashboard. The amount of previous weather and environment data that can be viewed by a user will be determined by the user with the options of 1 month, 3 months, 6 months and 1 year. The type of weather data to show will also be chosen by the user with the options being tree pollen count, grass pollen count, weed pollen count, temperature, humidity, UV index and air quality index. The data points on the graph will be daily averages for every day in the previously determined amount of time. This will allow users to see trends in weather data for a specific location over time and determine what times of year are the highest risk to the user. Weather data and pollen data will be provided by a Weather API and Pollen Count API. The API call to the weather and pollen API's will be done in one hit. A single call of all the data for the previous year will be made and the data displayed will be filtered client side. This is to avoid making unnecessary API call's and slowing down the rate at which data is displayed within the application. API call's will be cached on every visit to the application to prevent unnecessary large calls for weather data to the weather and pollen API.

## F8

A weather forecast for the next week will be provided within a widget on the main dashboard. The weather forecast will be displayed as a graph with icons showing the expected weather condition for each day. On hovering over the icons, the average data for the day will be shown as a popup showing precipitation, temperature, humidity, UV index and text description of the day where the cursor lands. On clicking the icons, the data will be expanded to show the hourly average data for the day. The icons will follow a simple convention of a cloudy icon, partly cloudy icon, rain icon, sun icon, snow icon, hail icon and storm icon. The data for the weather forecast will be provided via a Weather API. Clicking the hours on the expanded version of the weather forecast will display the average for that hour below the icons.

## F9

Air quality for the next 7 days will be provided within a widget on the main dashboard. For each day an icon will be shown for the tree pollen, weed pollen, grass pollen and overall air quality. The colour of each icon will determine the severity of the air quality from green (good) to yellow (mediocre) to red (poor) to dark red (very poor). On hovering over each pollen count icon, the pollen type and the ppm will be displayed. On hovering over the general air quality icon, the air quality index will be shown as a popup. On clicking one of the icons a graph showing the averages for the pollen type/air quality selected on an hourly basis for the selected day will be shown as a line graph.

Air quality and pollen count will be accessed via a Pollen Count API. The determined colour will be based on the ppm of each pollen count for the pollen data and the air quality index for the air quality. The ranges for tree pollen will be 0-95ppm for green, 96-207ppm for yellow, 208-703ppm for red and 704+ppm for dark red. The ranges for grass pollen will be 0-29ppm for green, 30-60ppm for yellow, 61-341ppm for red and 342+ for dark red. The ranges for weed pollen will be 0-20ppm for green, 21-77ppm for yellow, 78-266ppm for red and 267+ppm for dark red. The ranges for air quality index will be 1-3 for green, 4-6 for yellow, 7-9 for red and 10 for dark red.

## F10

A link to relevant articles about current weather conditions will be provided on the weather forecast widget and the air quality widget. Links will only appear in the case of extreme weather conditions. The potential weather conditions that will provide an article will be extreme temperatures, high air quality index, high pollen count. The articles linked will provide advice on how to cope with the weather conditions.

A link will be displayed for coping with moderately high temperatures when the apparent temperature for a location on a given day is >= 32 degrees Celsius and <= 40 degrees Celsius. A link will be displayed for coping with extreme high temperatures when the apparent temperature for a location on a given day is > 40 degrees Celsius. Apparent temperature will be provided by the Weather API as ‘feels like’.

A link will be displayed for coping with moderately low temperatures when the average wind chill temperature for a location on a given day is <= 0.5 degrees Celsius and >= -10 degrees Celsius. A link will be displayed for coping with extreme low temperatures when the average wind chill temperature for a location on a given day is < -10 degrees Celsius. The average wind chill temperature for a location will be calculated via the formula (average temperature (Celsius) – (average wind speed (mph) * 0.7)).

A link will be displayed for coping with moderately high tree pollen count when tree pollen count at a location on a given day is >=208ppm and <=703ppm. A link will be displayed for coping with extremely high tree pollen count when the tree pollen count is >703ppm.

A link will be displayed for coping with moderately high grass pollen count when grass pollen count at a location on a given day is >=61ppm and <=341ppm. A link will be displayed for coping with extremely high grass pollen count when grass pollen count at a location on a given day is >341ppm.

A link will be displayed for coping with moderately high weed pollen count when weed pollen count at a location on a given day is >=78ppm and <=266ppm. A link will be displayed for coping with extremely high weed pollen count when weed pollen count at a location on a given day is >267ppm.

A link will be displayed for coping with moderately poor air quality when the air quality index at a location on a given day is >=7 and <=9. A link will be displayed for coping with extremely poor ait quality when the air quality index at a location on a given day is 10.

### F11

Screen reader accessibility will be ensured by adding alt text to all image elements in the application. Headers in the application will be descriptive and concise. Punctuation throughout the application will be checked to ensure that screen readers can read the content with proper grammar.

## 2.3 Non-Functional Requirements Table

| ID | Requirement | Mandatory (Y/N) |
|---|---|---|
| NF1 | Average page rendering speeds below 1 second. | Y |
| NF2 | Average API call speeds of below 0.5 seconds. | Y |
| NF3 | The product must be compatible with Chrome, Mozilla Fire Fox, Safari, Internet Explorer, Opera and Microsoft Edge. | Y |
| NF4 | The product must be able to be accessed on any device with internet access and a browser. e.g., phones, computers, consoles etc. | Y |
| NF5 | The product must be written in UK English and contain no grammatical or spelling errors. | Y |
| NF6 | Value measurements will follow British conventions e.g., British Imperial for speeds and Metric for temperatures. | Y |
| NF7 | The product must be available to access 24 hours per day. | Y |
| NF8 | Navigation between pages must be available in a header bar for large screens and a hamburger collapsible sidebar for small screens easily seen by a user | Y |
| NF9 | Screen readers must be able to read all text and describe all images used within the software. | Y |

| | | |
|---|---|---|
| NF10 | Colour contrast for branding and headings must follow WCAG AAA guidelines. | Y |
| NF11 | Colour contrast for content and text in the application must follow WCAG AA guidelines. | Y |

## 2.4 Decomposition of Non-Functional Requirements

### NF1

Average page rendering speeds must be kept below 1 second. This is to ensure a quick and pleasant user experience. Average rendering speeds will be kept below 1 second by statically generating pages wherever possible so that they don't have to be rendered client side making rendering extremely fast. For pages that cannot be statically generated i.e., when a page contains dynamic data, API calls will be cached, and filtering of API data will be done client side to prevent waiting for API calls to complete taking a long time. Content on pages will be expandable when large amounts of elements are required to be rendered so that different sections of the page can be rendered as the user requires it. This will prevent long rendering times as users will only render information they require, and items will be rendered over time separately.

### NF2

Average API call speeds must be kept below 0.5 seconds. This is to ensure that data is returning quickly from API's and user's do not have to wait long periods of time to view data. API calls will be cached when a user makes one for an appropriate amount of time to be determined within the context by the developer. For example, a weather API call can be cached for a few hours as the data will not change rapidly over time. API calls will also avoid making large amounts of adjustment before being returned the client. Formatting API data if required will be done client side.

### NF3

Modern day browsers will be focused on for the development as legacy browsers are not in high demand. These will not need to be thought about much as modern day browses often support all the same features. The application will be tested on different browsers namely Chrome, Safari and Mozilla Fire Fox to ensure all features work efficiently and any issues that arise will be dealt with on a case-by-case basis.

### NF4

Responsive designs will be implemented to ensure all screen-sizes are able to be used for the application. Hamburger menus will be used that open a collapsible side bar for navigation when small screen sizes are being used to allow for full navigation with ease by the user when a nav-bar is not appropriate for the screen size. Touch controls will be checked on all aspects of the application to ensure all features can be used effectively by users. Landscape UI features such as the daily weather forecast widget will be made into a portrait list to prevent side scrolling on small screens. Graphs will be scaled down. Issues with clicking graphs will be a limitation of the application on small screen however, dots will be provided to indicate where the graphs are clickable.

### NF5

The developers will be fully proficient in UK English, and grammar will be peer checked by an AI and a reviewer to ensure that all grammar is correct.

### NF6

Developers will be made aware to ensure that all values are measured in British conventions and all measurements will be clearly labelled to ensure that the user is aware of the measurement system being used.

### NF7

The product will be deployed on servers that are running 24/7 and backup servers will be ensured so that if any issues occur on one server the application is able to start back up on another server as soon as possible.

### NF8

Navigation must be clearly displayed to users at the top of each page so that users can access the navigation bar with ease. Labels will be descriptive and short to ensure that users can get a basic understanding of what each page is without needing to click into them.

### NF9

All images and text must be easily readable by screen readers. Images will be provided with alt text in their element to ensure that all images can be described by a screen reader. All text on the application will be checked for proper grammar and punctuation to avoid confusion with phrasing when using a screen reader.

### NF10

Header text and branding text must be in a 14pt or higher font size, bold and have a contrast ratio of at least 4:5:1 to ensure AAA WCAG Guidelines are follow for better accessibility of headers. Branding images and graphical objects must have a contrast ratio of at least 3:1 so that the branding is clearly visible at different sizes.

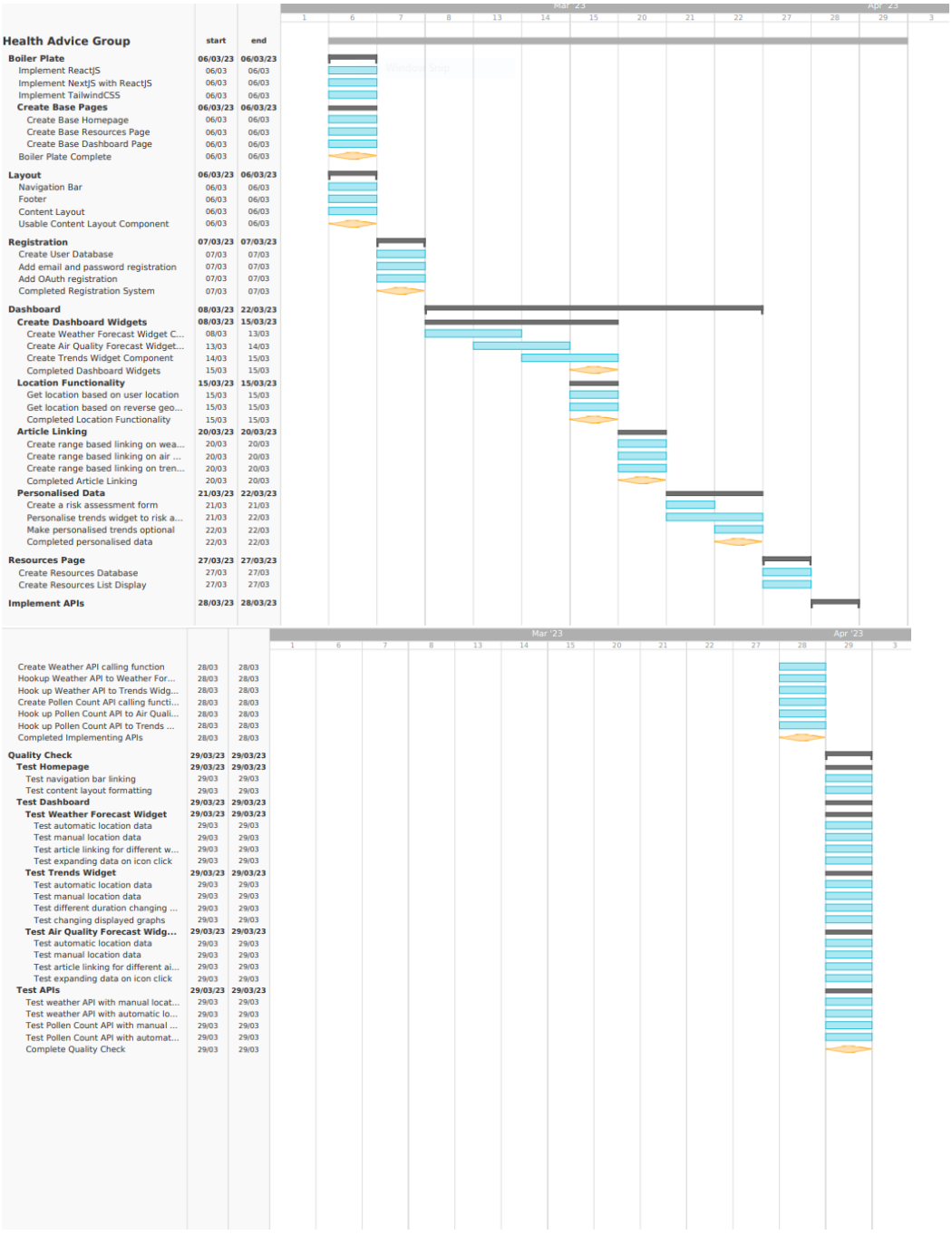
### NF11

Content text must be in a 9pt or higher font size and have a contrast ratio of at least 4:5:1. This will be primarily focused on text within blocks of text. User interface components must have a contrast ratio of 3:1 and graphical objects must either have a contrast of 3:1 or descriptive text with a contrast ratio of 4:5:1.

# 3.0 KPIs and User Acceptance Criteria

## 3.1 KPIs

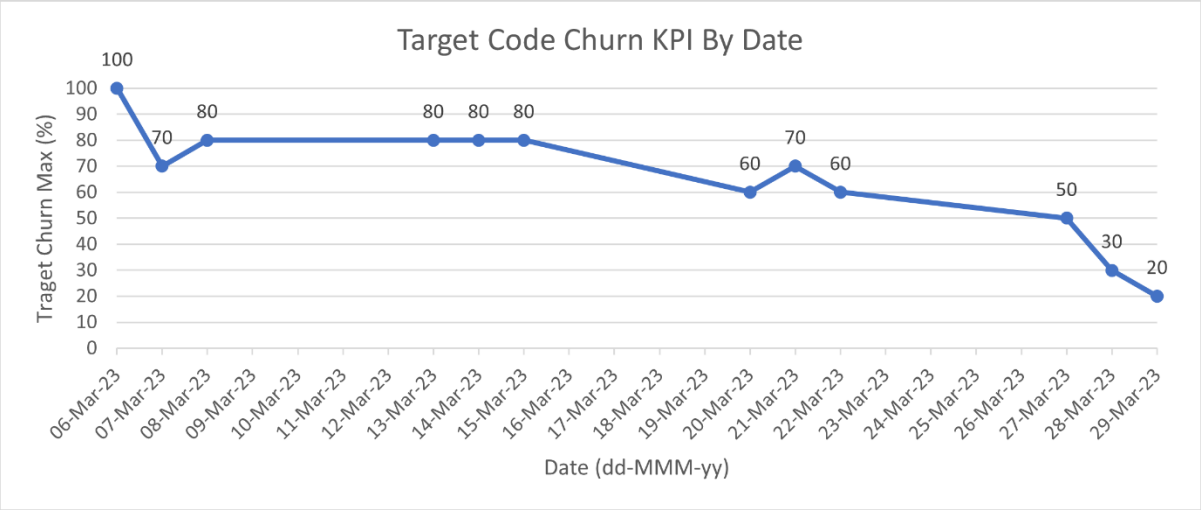
### Gantt Chart




*A Gantt Chart displaying the target dates for development completion and milestone dates within the project*

A Gantt chart will be used to determine whether the project remains on progress throughout its development and to measure the priorities of the project as development continues.

## Target Code Churn KPI By Date




*Target Code Churn KPI displaying the maximum code churn percentage targets for a given date within the project.*

Code churn will be measured throughout the project with the maximum code churn targets being stated in the graph displayed above. Target maximum code churn percentages are determined by the amount of functionality that needs to be developed on the day based on the Gantt Chart displayed above.

## Cumulative Flow KPI

Cumulative flow will be measured throughout the project to ensure that the number of tasks that are being completed per day is on track with the number of tasks expected to be completed per day. The number of tasks expected per day will be determined by the Gantt chart and task manager that will be used in tandem with the Gantt chart. If the gap between the accumulated tasks and the number of tasks completed begins to widen a reordering of priorities will need to occur to ensure all core functionalities of the project are competed in time.

## Defect Density KPI

Defect density will be measured throughout the project to ensure that the codebase continues to be rigid, and the number of bugs does not become too high. Defect density will be kept below 0.01 as a target throughout the project.

## Sprint Burndown

Sprint burndown will be measured throughout the project to ensure that sprints are completing the number of tasks required and that the amount of effort required is not overbearing for the development team.

## 3.2 User Acceptance Criteria

| User Requirement | Acceptance Criteria |
|---|---|
| A user must be able to see a weather forecast to inform on health decisions. | - A widget that displays the next 7 days weather forecast on a dashboard<br>- A link to an article explaining relevant health information about current weather conditions<br>- Ability to expand the weather forecast widget with individual days hourly forecasts |
| A user must be able to see an air quality forecast to inform on health decisions. | - A widget that displays the next 7 days air quality forecast on a dashboard<br>- A link to an article explaining relevant health about current air quality conditions<br>- Information on tree pollen ppm displayed in the widget<br>- Information on weed pollen ppm displayed on the widget<br>- Information on grass pollen ppm displayed on the widget<br>- Information on air quality index displayed on the widget |
| A user must be able to get advice on how to manage health matters related to weather and environmental conditions. | - A resource page listing all information articles<br>- Articles with accurate and correct information that can be accessed through the application by a user<br>- Linking to relevant resources when weather conditions or environmental conditions on the dashboard are of concern via the dashboard widget |
| A user would like to have personalised health advice based on location. | - Ability to automatically detect a user's location if the user permits<br>- Ability to enter a manual location to measure the weather and environmental conditions at that location<br>- Linking to relevant health articles based on weather conditions in inputted locations via dashboard widgets |
| A user would like to have accessibility features to ensure that they can use the application regardless of physical ability. | - Screen reader compatibility throughout the application<br>- Accurate labelling of images using alt tags in HTML<br>- AAA WCAG guidelines conformity on text for headers and branding<br>- AA WCAG guidelines conformity on text for content and graphical objects |

| | |
|---|---|
| A user would like to see a tracking tool for weather and environment conditions to be able to track personal health risks in different environments and weather. | - A trends widget on the dashboard showing average values for a certain amount of previous duration<br>- Ability to see trends from 1 year previously to current time<br>- Ability to see trends from 6 months previously to current time<br>- Ability to see trends from 3 months previously to current time<br>- Ability to see trends from 1 month previously to current time<br>- A graph showing the average tree pollen daily for the duration selected<br>- A graph showing the average weed pollen daily for the duration selected<br>- A graph showing the average grass pollen daily for the duration selected<br>- A graph showing the average air quality index daily for the duration selected<br>- A graph showing the average temperature daily for the duration selected<br>- A graph showing the average UV index daily for the duration selected<br>- A graph showing the average humidity daily for the duration selected<br>- Ability to select which graph is visible |

# 4.0 Proposed Solution Description

## Proposal Overview

The Health Advice Group project is a project to develop a digital solution that can be used to forecast weather, monitor air quality data and get advice on health conditions affected by weather and environmental conditions. The Health Advice Group project will be developed in JavaScript, HTML, CSS and SQL. The final solution will be a web application accessible through standard modern browsers e.g., Google Chrome, Mozilla Firefox, Safari etc. The Health Advice Group project will be developed using the ReactJS framework in Next 13. The timeframe provided for the Health Advice Group project is 4 weeks with 30 working hours.

## Proposed Solution Scope

The Health Advice Group project solution will have a homepage whereby a basic dashboard will be accessible containing the base weather forecast widget and an air quality monitoring widget. The homepage will also contain links to recent articles added to the database that can be navigated to directly from the homepage. The homepage will display the 3 most recent articles and only the weather forecast widget and air quality widget.

The Health Advice Group project solution will have a dashboard page with more extensive weather and environmental information. The dashboard page will contain a weather forecast widget, an air quality monitoring widget, a trends widget and linking to relevant environmental articles via the dashboard widgets.

A user will be able to login to the web application either through email and password or OAuth. Logged in users will be able to keep personalised data to customise their experience in the web application. Personalised data will include relevant health conditions to the user, default location data and preferred widget graph defaults.

Location data will be used to determine weather and air quality data for the user. Location data will either be acquired automatically by the browser if the user grants permission for location data to be collected or will be manually inputted by the user. Manual location data will be able to be inputted via a text input box that accepts any form of address or location. Upon searching for a location, the user input will be sent to a Reverse Geocoding API where the coordinates of the inputted location are determined.

The weather forecast widget will show the current forecast for the next 7 days based on the location data collected by the application, if no location data is present the weather forecast widget will default to showing the data in London, UK. If a user is logged in a default location may be selected which will be used instead of the London, UK default when no location data is provided.

The weather forecast widget will display the 7 days with icons that can be clicked to change the widget to display the day in hourly breakdowns of averages. Underneath each icon will be the average precipitation, UV index, temperature and humidity for that day or hour depending on the data selected to be shown. The icon will be based on the general weather conditions for the day e.g., a rainy cloud if there will be mostly rain and clouds on a given day or hour. On clicking an icon, the hourly averages for the clicked day will be shown in the same format as the 7-day breakdown screen. The user will be able to return to the 7-day forecast widget screen by clicking a back button in the top left of the widget. All weather forecast data will be pulled from an external weather API.

The weather forecast widget will provide linking to relevant articles based on current weather conditions. Linking will only display in the event of moderately high – extreme weather conditions or abnormal or potentially dangerous weather events. For example, a moderately high – extreme weather condition such as high temperatures will link an article to coping with hot temperatures. An abnormal or potentially dangerous weather event such as a lightning storm will also link an article explaining how to stay safe during the weather event.

The air quality monitoring widget will show the current forecast for the next 7 days based on the location data collected by the application, if no location data is present the air quality monitoring widget will default to showing the data in London, UK. If a user is logged in a default location may be selected which will be used instead of the London, UK default when no location data is provided.

The air quality monitoring widget will show a forecast of the air quality for the next 7 days. Each day will have 4 icons a tree, a weed, a tuft of grass and a wind icon. Each icon will be labelled. The tree icon will be labelled ‘Tree Pollen’, the weed icon will be labelled ‘Weed Pollen’, the tuft of grass will be labelled ‘Grass Pollen’, and the wind icon will be labelled ‘Air Quality’. On hovering over a pollen icon, the average ppm for that pollen on that day will be displayed. On hovering over the air quality icon, the average air quality index for that day will be displayed. On clicking an icon, the hourly averages of the day clicked for the selected air quality type will be displayed in the widget as a line graph. For example, clicking the weed pollen icon on the 7th of February 2023 will show a line graph of the hourly weed pollen averages for the 7th of February 2023. The user will be able to navigate back to the main 7-day forecast widget screen by clicking a back button in the top left of the widget. All pollen and air quality data will be pulled from an external pollen count API.

The air quality monitoring widget will provide linking to relevant articles based on current air quality conditions. Linking will only display in the event of moderately poor – extremely poor air quality conditions. Links will appear under each icon as a ‘See Advice on Conditions’ label that can be clicked to see a relevant resource.

The trends widget will show previous daily averages for a selected time frame based on the location data collected by the application, if no location data is present the trends widget will default to showing the data in London, UK. If a user is logged in a default location may be selected which will be used instead of the London, UK default when no location data is provided.

The trends widget will show data based on a selection made by a user as a dropdown selection at the top-middle of the widget. The options for the data to view will be tree pollen count, grass pollen count, weed pollen count, temperature, humidity, UV Index and air quality index. The data shown will be previously collected daily averages for the given location. The amount of time the data goes back will be determined by the user. A dropdown in the top right corner of the widget will allow the user to select between 1 month, 3 months, 6 months and 1 year. The default values will be temperature for a duration of 3 months. All data will be shown on a line graph with points at each day that can be hovered over to see the average value for the selected data type on the day hovered over.

The Health Advice Group Project will have a resources page with a list of all the resources available within the application. There will also be a search bar to search all the resources for a specific resource. The articles will be grouped into different sections with a relevant header for example temperature conditions, air quality conditions etc. Each section will be collapsible to hide the articles within that section. All sections will default to collapsed on entering the resources page. On clicking a resource, it will navigate the user to the resource for viewing.

## Proposed Timeline and Milestones

The Health Advice Group Project will be developed over 4 weeks between the dates 6th of March 2023 to 29th of March 2023. The milestones and their targeted dates for the project are as follows:

- Boiler Plate Completion – 06/Mar/2023
- Usable Content Layout Completion – 06/Mar/2023
- Completed Registration System – 07/Mar/2023
- Completed Weather Forecast Widget – 13/Mar/2023
- Completed Air Quality Monitoring Widget – 14/Mar/2023
- Completed Trends Widget – 15/Mar/2023
- Completed Location Functionality (manual and automatic) - 15/Mar/2023
- Completed Article Linking – 20/Mar/2023
- Completed Personalised Data – 22/Mar/2023
- Completed Resources Page – 27/Mar/2023
- Completed API Implementation – 28/Mar/2023
- Completed Quality Check – 29/Mar/2023
- Deployment – 29/Mar/2023
- Project Completion – 29/Mar/2023

# 5.0 Justification

## 5.1 Justification of Proposed Solution in Relation to Client and User Needs

The Health Advice Group project will be built in JavaScript, HTML, CSS and SQL. The frameworks used will be ReactJS in Next 13. Next 13 will be used as it provides static site generation meaning pages can be rendered on the building of the codebase. Therefore, accessing static sites is very rapid and will provide a good user experience. Static site generation only applies to pages where dynamic data is not required however, so it will only be applied to the resources page. This will however make navigation around the resources page very quick and allow for users to access necessary resources and advice with minimal delay.

ReactJS is a very widely used framework for developing modern day web applications. ReactJS allows for custom components to be developed that can be implemented into files with ease. It will allow for clean code creation and avoiding large files that are difficult to read. ReactJS is very expandable and will allow for future developments to occur with ease whenever the client requires further updates.

Tailwind CSS will be used to develop the CSS for the application. Tailwind CSS is a widespread CSS framework that can be used to speed up CSS development. Classes are not required to be made manually in a CSS file for most situations and readability of the code is significant increased as all the CSS information required is visible in the same file as the component the CSS is being applied to. This will allow for easier understanding of the code base by future developers when updates are required by the client.

The homepage containing a simplified version of the dashboard page allows for users to access basic information with ease when in depth reviewing of information is not required. It will allow users to see the next 7-days weather forecast and the next 7-days air quality averages without having to navigate the web application. This will increase user experience and increase customer retention as customers will find the web application more pleasurable to use.

The homepage weather and air quality monitoring widgets will not display hourly averages for a given day when clicked, instead it will navigate the user to the dashboard. This is to decrease the amount of time required to render the homepage as less API data will be required to be called. This will also reduce strain on the API ensuring that the API calls are kept within 0.5 seconds. A fast-loading homepage is also necessary to not frustrate users when they do not require access to API data but are instead possibly navigating to the web application solely for access to the resources.

Recent articles will be displayed on the homepage as it will effectively create a ‘news’ section for the application. Having a section that is known to consistently update with new resources and articles to read will increase the amount that users check the web application. Therefore, the web application traffic will increase. This is because users will have a reason to return to the web application frequently. Therefore, the client will benefit from the increased web application traffic and users will benefit from ease of access to recent articles and resources that may prove interesting to them.

The dashboard page will contain 3 widgets that can be used by the user. The weather forecast widget, air quality monitoring widget and the trends widget. The weather forecast widget will display more in-depth data than the homepage widget. It will provide linking to relevant articles and be expandable to display the weather data for individual days on an hourly basis. The weather forecast widget will provide the customers with their expected service of being able to see the next 7-days forecast. The forecast is limited to the next 7-days as any further will not be relevant to the customer in the context of the application. The application is intended to display upcoming potential health risks and not for planning far in the future. A 7-day limit also ensures the most accurate data as data will not be based as heavily on estimations. This is an important factor for providing health advice as the application will want to ensure it is providing data as accurately as possible.

The weather forecast widget will be able to be expanded to a daily overview with averages on an hourly basis. This is so that users can get an accurate representation of when health advice is required on a given day and when health advice is not necessary to be followed. An hourly basis was chosen as it provides a good timeline of weather conditions throughout the day without overloading the user with information that is not necessary say if a minute-by-minute basis was chosen instead.

The air quality monitoring widget will be displaying a forecast of the next 7-days showing the tree pollen count, weed pollen count, grass pollen count and air quality index for each day. A 7-day limit was decided as the application is intended to display upcoming potential health and not for planning far in the future. A 7-day limit also ensures all data is accurate an important characteristic when providing health advice to users. The types of pollen selected were based on the most common types of bothersome pollens for people with environmentally related health conditions. Tree pollen, grass pollen and weed pollen often are the ones that trigger hay-fever and are most abundantly found across the world. Air quality index was also selected as it is a good indication of the potential for health risks related to the environment. It accounts for allergens, pollution and particulates in the air therefore, giving a good overview of potential dangers.

The air quality monitoring widget will allow for expanding the pollen count / air quality index of a day to an hourly report. The hourly report will display its data in a line graph as pollen count changes rapidly and an indication of when the air quality is increasing or decreasing over a given time frame is a good way for users to plan their days according to air quality. An hourly basis was decided as it does not overload the user with information and provides enough information to interpolate the data between two different hours with a decently high degree of accuracy. The air quality monitoring widget fulfils the ‘access to a dashboard for monitoring air quality data’ requirement from the client.

The trends widget will display previous data within a set time frame selected by the user on different types of data also selected by the user. The options for data to display will be tree pollen count, weed pollen count, grass pollen count, UV index, humidity, temperature and air quality index. The selected data types to allow the user to choose from was decided by their significance in providing health advice for users. Options such as precipitation are not as relevant when determining the health risks for a person. The trends widget will provide a method for users to track their health risks in different locations and see how they need to prepare for different environments.

The trends widget will have the option to show previous data in the time frames 1 month, 3 months, 6 months and 1 year. These time frames were decided as they can provide current weather and environment trends (1 month), seasonal weather and environment trends (3 months), seasonal change weather trends (6 months) and general trends of weather and environment changes throughout a year (1 year). Choosing a decrease of less than 1 month appeared redundant as the forecast widget can be used to see very current weather trends and a choose a time frame of over 1 year appeared redundant as the application is not for research purposes but for receiving health advice. The API call strain on a call over 1 year would be much greater as well and would decrease the speed of the application and the potential costs of running the application. Therefore, the selected time frames were decided upon.

The resources page will display all the resources within the application in collapsible groups with a search bar available to search all resources. The resources list was decided to be grouped and collapsible as it allows for easier navigation of all resources when a user requires a specific resource. Collapsible groups will also allow for a faster rendering time on the resources page.

The resources page will be statically generated as data will not need to be dynamic. The drawback of statically generating the resources page will be that a new build must run every time new resources are implemented however, the benefits of having a very rapid resources page outweigh this drawback. User experience will be significantly increased by a rapid resources page that can open articles and resources quickly.

The resources page search bar was decided to be implemented as another method of increasing user experience as it will make it easier for users to search for specific resources. The search bar will allow for searching for groups and showing all resources within a searched group and specific resource names. This is to further increase the user experience and make user retention higher. Therefore, benefitting the customer and the client.

The application registration system will provide the option for both OAuth and manual email and password registration. This was decided as not all users will have a google account therefore, they can use their own email accounts for the application. However, the users who do have google accounts will have a very easy and quick registration and login experience. Security will also be handled by OAuth which is renowned for being very secure and will mitigate the risk of a security breach.

SQL will be the language used to develop the databased for the application. Specifically, PostgreSQL as it provides JSON support. JavaScript objects are designed to be structured following the same conventions of JSON therefore, a database using JSON objects will be very compatible with the application. SQL is also very rapid, and databases can be built quickly with efficiency and cheaply.

Screen reader accessibility will be implemented and tested throughout the application. This is to ensure that the application is accessible to people hard of sight. The colour schemes and the text sizes will also be designed to follow AAA WCAG guidelines for branding and headers and AA WCAG Guidelines for content. AA WCAG was decided for the content as it allows for more freedom with creativity when designing the website. However, AAA WCAG Guidelines for headers and branding was decided to increase the visibility of branding and headers regardless of size. For example, branding in a favicon will need to follow AAA guidelines as it will need to be visible at very small sizes.

## 5.2 Risks and Mitigation

A potential risk for the application is a security breach which could lead to a leak of sensitive personal data. This risk will be mitigated by using OAuth and a fully encrypted user database. OAuth is well known for being very secure and providing all the authentication needed to keep a database safe out of the box. An encrypted user database with hashed passwords will allow for protection against hackers attempting to view the user database.

Another potential risk for the application is inaccurate forecasting of weather and air quality data. Forecasts cannot always be fully accurate therefore the user needs to be made aware of the potential for inaccuracies. This risk will be mitigated by limiting the time frame of forecasting to 7 days. This will be a relatively accurate time frame as there will be less guess work and interpolation used to determine the forecast.

Inaccurate resources could pose a risk to the application as inaccurate resources could provide users with inaccurate medical advice. This could pose legal risks for the client. Mitigation of inaccurate resources will have to primarily be carried out by the client however, it can be done by using peer reviewing of resources added to the application before they are added to the database. Therefore, all resources will be quality checked to ensure accurate information.

A risk potentially faced by the application is the failure of servers that the application is deployed to. This risk can be mitigated by using a third party well accredited hosting provided that provides guarantees on movement to different servers if a server an application is deployed to happens to fail. A backup deployment of the application to another server cannot also be used to mitigate the down time of a server failure by triggering a deployment again the moment a server that is deployed to goes down.

## 5.3 Regulator Guidelines and Legal Requirements

The Health Advice Group Project solution will be deployed to servers in the UK and the EU. Therefore, it will follow GDPR guidelines on storing data. This means that it must not be deployed to the US or any other international territories outside of the UK and EU until the data storage methods have been checked to match the local data laws.

The Health Advice Group Project will be providing medical advice however, on forecasts the advice provided cannot be guaranteed to be entirely accurate as forecasts are not always entirely accurate. Therefore, the solution must provide a clear notice to users that the advice provided should not be taken as medical advice and if they have concerns, they should seek professional help rather than rely on the solution for answers.

The Health Advice Group Project will store personal and sensitive data on user's locations, health conditions and personal details. Therefore, the application must provide a notice to users that these details will be stored, and the user must approve the storage of this data before they may proceed with using the application. This only applies to logged in users.

Location data will be taken from user's current location. A notice must be made to the user before location data is taken to ensure their approval of the use of their location data. Cookies may also be used with the user to store personalised information on the user's use cases within the application. A notice must be given to the user about their approval of using cookies and no cookies can be used on the user if permission is denied.