Problem Specification – Causeway Coast & Glens Interactive Web Mapping (GIS)

**Project Brief** – Using interactive web-based mapping, develop a range of features to support the development of a BeachSmart campaign informing residents & visitors of environmental issues they should be aware of before using the renowned beaches & coast of the Causeway Coast and Glens.

**Problem Domain**

* In 2019, the Royal National Lifeboat Institution (RNLI) attended 17,000 incidents, giving aid to nearly 30,000 people. Of these, 200 alone were along the coast of Northern Ireland(<https://www.communityad.co.uk/rnli-hm-coastguard-launch-beach-safety-campaign-urging-parents-protect-families-save-lives-coast-summer/>).
* As a result, following the easing of lockdown restrictions in May 2020, fearing an influx of visitors to the coast, the RNLI and HM Coastguard launched a “beach smart” campaign (<https://www.northernirelandworld.com/news/new-rnli-and-hm-coastguard-campaign-2860751>), aiming to inform residents and visitors on how to enjoy the beaches and coastline of Northern Ireland, England, Wales and Scotland safely and with consideration of the natural environment. This was done in the hopes of reducing the load on their, already strained, resources.
* Comprising 12 district councils (<https://www.ark.ac.uk/elections/nlgccg.htm>), Coast & Glens Council (CCG) oversees 11 beaches stretching from Benone to Cushendall (<https://www.causewaycoastandglens.gov.uk/see-do/beaches>).
* There is limited, in-depth information, that is easily accessible in one place online about the coast in general but the beaches. This makes it difficult, particularly for elderly and people with disabilities, to access to required information to enjoy the coastal features safely.
* It is hoped that by making this information easily accessible to more people, there will be fewer incidents requiring the attention of the RNLI or coastguard.

**Addressing the Problem**

* The beach smart web app aims to provide a single portal for all things CCG. Users will be able to use the app to explore CCG beaches, coastline and towns, including local landmarks, entertainment, eateries and shops.
* This, primarily, is aimed at improved beach safety education among visitors, local and otherwise. Having said that, pre-pandemic there were 5.3m tourists visiting Northern Ireland, this figure (<https://www.tourismni.com/industry-insights/tourism-performance-statistics/>). The COVID-19 pandemic saw significantly fewer tourists visit. Given that pre-pandemic tourism contributed £1bn to the NI economy (<https://www.economy-ni.gov.uk/topics/tourism#:~:text=NI%20Annual%20Tourism%20Performance%202019,-The%20latest%20Northern&text=The%20NISRA%20report%20points%20towards,spend%20when%20compared%20to%202018>), of which 70% was from external visitors, it is important to be rich in functionality that applies to tourists as well as locals, to take advantage of recovering tourism numbers. This could have a significant impact on the economic prosperity of the area in the coming years.

**Proposed Product Features**

*MoSCoW Prioritisation*

* Must have
  + Accessibility features for users with keyboard-only or visual/auditory impairments – to include light/dark themes and high contrast mode as well as support for text-to-speech services.
  + An interactive map with pins displaying the beaches that the council oversees.
  + Users must be able to explore the map, to include zooming and panning.
  + A link for each beach to the relevant CCG website page.
  + Environment, historical and practical (accessibility, closure times etc.) information for a beach that is selected. To include beach status such as flag and water category.
* Should have
  + Local landmarks, businesses and sports facilities, highlighted with icons/pins – to include opening hours, contact details and links to external websites.
  + A map filter allowing users to decide what type of amenity or attraction type they wish to browse.
  + Live weather and tidal data allowing users to safely explore the area.
  + Bus and train routes around the area with timetables (better UX) or links to external Translink website.
  + Users should be able to create an account – allowing them to add map items to a watchlist, get notifications about events and save favourite routes – could sign up with email or Google account.
* Could have
  + Users could pick a list of beaches, landmarks etc they wish to visit and give their available time, start and end point. An algorithm would be used to calculate their most efficient route. This is perhaps a more tourist-oriented requirement but could be of great use to locals too.
  + Users could report any rubbish, damaged property or anti-social behaviour to the council, perhaps uploading a photo from the app. This would be reported along with time, date and geolocation data, allowing the council to take quick and efficient action.
  + Users could report sightings of wildlife e.g., birds, whales, dolphins or seals, which occasionally show up on the North Coast. These could be pinned to a location to let others know where they may catch a glimpse. This could be used also to highlight areas where people have, for example, been stung by jellyfish. This could allow others to bathe and enjoy the beach safely and with greater peace of mind.
  + An API, likely built using PHP or NodeJS, allowing other developers to make use of the aggregated data used in the app.
  + Social media aggregators which display tweets or Facebook posts using pre-determined hashtags or the CCG social feeds, as per “plan your trip” page on the CCG website (<https://www.visitcausewaycoastandglens.com/plan-your-trip/visitor-information/maps-and-guides>).
* Won’t have (within the scope and time-horizon of this project)
  + Integration with Fitbit, Strava or similar, to save routes or walks and share progress with friends and family.
  + Social media integration allowing posts to be made direct from the web app.
  + Photos uploaded by viewers aggregated together and shown in the information section of each pin location as a slideshow with username and date below.
  + Augmented reality functionality, allowing users to preview the area and decide as to where they wish to visit.

**Technology Considerations**

This can be seen as an initial plan. First client meeting is not scheduled at the time of submission, therefore all technology and architecture considerations are subject to change over the course of the project, as is the norm, dependent on the Council’s current set up and how tightly integrated they wish it to be.

*Front-End*

* HTML 5 used for structure of pages.
* CSS – Bootstrap 5
  + Bootstrap to be used for a mobile-first build which will scale on desktop. As this service will be most useful for users when out and about it is important that the functionality within reflects this use-case. Bootstrap allows for a lightweight, responsive design with components that are easily customisable and offer high levels of functionality.
  + Allows access to open-source icons for customising the look of the site. As these come with Bootstrap, they are free to use without additional accreditation, reducing the likelihood of legal complications arising from the finished product.
* CSS – General code design and UX considerations
  + Global variables to be used where possible for colours, fonts and any other constant values that are to be reused throughout the codebase.
  + Height, width, margins, etc., to be controlled vw and vh, keeping them relative to the width and height of the given viewport. This will be important for maintaining responsiveness across an array of device screen dimensions.
  + Likewise, font sizing to be controlled using relative units – rem or em.
  + Font family: Arial, Helvetica, sans-serif – gathered from CCG website using Font Finder chrome extension. A simple, clean and crisp font-family that is easily recognisable. Scales well as device size changes.

Graphical user interface, application

Description automatically generated with medium confidence

* + Logo – there doesn’t seem to be a beach smart logo as such. As this is a joint venture between CCG and the RNLI, these could be used. If allowed, having the opportunity to design a logo would be an exciting prospect.
  + At this point there are no definitive requirements for design aspects so initial thoughts are with the current colour palette for CCG site in mind. Personally, I think the CCG colour scheme isn’t particularly modern but will be used if the client wishes to keep consistency across products. If given some freedom in this decision a different colour palette will be developed.
    - Site palette - Jet: #333333ff; English-violet: #503250ff; English-violet-2: #6a4c6dff; Quinacridone-magenta: #85425cff; White: #ffffffff; Dark-cornflower-blue: #163982ff; Blue-sapphire: #195d70ff
    - Extra colours which are complementary – palette created using <https://coolors.co>. Inclusion of blue colours will bring some contrast to the current, darker, CCG colour scheme. They also lend themselves
      * + Opal: #9bc4bcff; Aero-blue: #d3ffe9ff; Middle-blue: #8ddbe0ff

Chart, treemap chart

Description automatically generatedChart, treemap chart

Description automatically generated

* Mapping APIs (JS) – a couple of interactive mapping APIs have been considered, with the two main options being Google Maps and Open Street Maps
  + Google Maps
    - Widely adopted by all browsers, small learning curve for most users.
    - Highly customisable with good map controls.
    - Expensive to use at larger scale - $300 free trial mode, then $200 monthly credit for Google maps platform - <https://developers.google.com/maps/billing-and-pricing/billing#monthly-credit>.
  + Open Street Maps
    - Open-source, free to use mapping system. Community driven and maintained which gives the option of maintaining accuracy for more obscure localities. This gives the developer more control as the map can be edited to remove any inaccuracies.
    - Good API documentation available.
* Java Script
  + React or React Native for Front End UI
    - React is a Java Script library rather than framework, allows for the creation of reusable components for the app front end.
    - High speed and good performance due to the virtual DOM (<https://massivepixel.io/blog/react-advantages-disadvantages/>).
    - Allows for quick iterations of product and should make responding to client requests for edits more straight forward.
    - Search Engine Optimisation friendly (<https://www.javatpoint.com/pros-and-cons-of-react>), comes in handy for making people aware the product exists.
    - Updates are carried out regularly, keeping things up to date with modern standards.
  + Leaflet JS to be used for map integration.
    - Open-source framework that is designed to be mobile-friendly. As this project will be developed “mobile-first” this seems to be appropriate.
    - Works seamlessly with Open Street Maps, with an easy-to-follow guide for implementation.
    - Core features are designed to be extremely lightweight but there are several plug-ins available to add extra functionality.

*Back-End*

* Server-side
  + Node JS
    - Runtime environment for Java Script, allowing it to be used on the server-side.
    - Developing full stack in Java Script gives certain benefits such as high speed, good performance and good efficiency (<https://www.altexsoft.com/blog/engineering/the-good-and-the-bad-of-node-js-web-app-development/>).
    - Can use Express JS framework alongside, although many other options available too.
* Database
  + MySQL
    - Widely used legacy system with syntax I am somewhat familiar with already.
    - Secure Socket Layer makes transmitted data well protected.
  + Mongo DB
    - No SQL database – that is rather than a relational data system as with MySQL, Mongo DB is an object-based system that employs the use of JSON objects.
    - More flexible when it comes to searching for data and building a dataset.
  + Both are support Java Script server-side technology. MySQL seems more appropriate for high traffic sites that require high levels of security. Mongo DB is said to be more applicable for sites with an analytical focus.
  + As for which is better, MongoDB being part of the MERN stack containing React may be more appropriate, despite the steeper learning curve having never used it before.

Project Plan (GANTT)

Chart

Description automatically generated