**d18124788 - Liam McKenna**

**dt249 - Internet Application development Assignment**

**-**

**Table of Contents:**Overview.  
Goals.  
TECH STACK.  
HOSTING.  
INITIAL FEATURES.  
DESIGN/ Site Map.

**OVERVIEW:**An overview of the project can be detailed as a full-stack (aka a 3 tiered client/server architecture) website application. The web app will need to present statistical information related to the Covid19 virus in various countries in a front-end UI accessible to the public and store this data on a backend database server.

**goals:**

* Build a user-friendly web UI to allow clients to access the information they want and present the covid19 data in a easily readable format.
* Build searching and filtering tools to make it easy to find data.
* Build and host a database to store the data required for the Website UI.
* Initial instruction is to use a dataset provided by the lecturer, however intention is to find a suitable dataset API online that can be used to populate the database.
* Create a user signup and login system to access restricted pages. Initial instruction from lecturer is to use this to facilitate an administrator access to alter functions on the website, however further discussions have found that this may not be suitable. Instead, allowing clients to signup and login to the website to access extended data may be more suitable.
* Ensuring the website is secure from potential malicious attacks such as SQL injection so that user data remains safe and the data cannot be altered or destroyed by a unauthorised party.
* Ensure the web application is mobile friendly across multiple devices.

**Tech Stack:**XAMPP, PHP, HTML, CSS, Javascript, MySQL  
For this project, I intend to keep with the above technologies. Given the module has focused on these It would make sense to apply the knowledge that has been taught.  
  
Setting up of the developer environment will be with Visual Studio Code for the front end and Datagrip for the backend.  
  
I initially wanted to use newer adopted technologies such as Node.Js, React.Js, tailwind, Typescript, Sass and even thought about looking into even higher prototype tech such as Web Assembly with Rust. However, while this could cause a problem by diverging from the source material of the module, my main reason to focus on the classic web stack was to expand my knowledge on how a full stack web application will be developed with technologies like PHP and possibly understand why the industry has started to move away from this kind of tech stack to modern stacks such as MERN/MEAN.

**HOSTING:**Hosting the application I could possibly use the Microsoft Azure platform as this is the only cloud hosting system that’s been taught to us in another module. Or alternatively I think a more suitable hosting solution will be using Heroku as my research has found this to be better suited to PHP applications. This will require more research to confirm if this is a better hosting platform after the application is in a state ready to be hosted.

**INITIAL FEATURES:  
Database Connection to historical Data.**I’ll need to research how and create a connection from my front-end website to my backend database to allow MySQL queries to be performed and the data returned to my front-end website.

**Get Covid19 data from an API and post to database.**Initially I will be using a pre-made dataset fron an excel document, however I intend to move over to a public API to gather the Covid19 data. The lecturer has advised this process will be explained in a later lecture.

**Modular data preview Cards for each Country with a ‘see more’ option to get full details.**The data retrieved from the database will be organised and categorised by Country. The home view will have a ‘preview’ view of a limited amount of data for each country, and there will be an option to ‘see more’ to view the complete list of data on a new page. I intend to make the ‘full data’ page dynamically populated depending on what country was selected. This will ensure there’s not a unique page for every country and allows for reusable code.

**Data Charts and visual information.**I intend to have visually appealing charts and infographics visible in the countries pages to give a better understanding of the data associated with each country.

**Sideview Navigation bar.**I intend for the navigation bar to be a sidebar design to the left of the page for both desktop and tablet devices. This Decision came from the decision that vertical screen real-estate would be more beneficial on landscape sized screens. Also I find it to be more aesthetically appealing and modern looking,

**Mobile friendly Navigation bar.**I intend to pay attention to how the navigation tools will be located on mobile and portrait screens. Much like a lot of modern native mobile applications, the navigation tools are located at the bottom of the screen. This will make it easier for the users fingers to reach the applications navigation options while also being more aesthetically appealing.

**Page content paginator.**To make sure that there is not too many countries loading on the main page and to make sure that each country doesn’t load too much data, I will create a paginator system to only show a restricted amount of data and a page counter and option to move forward and backwards through the pages of data. This will help to stop the users page becoming way too cluttered and large, it will also prevent loading issues where there may be a large amount of data being retrieved and presented.

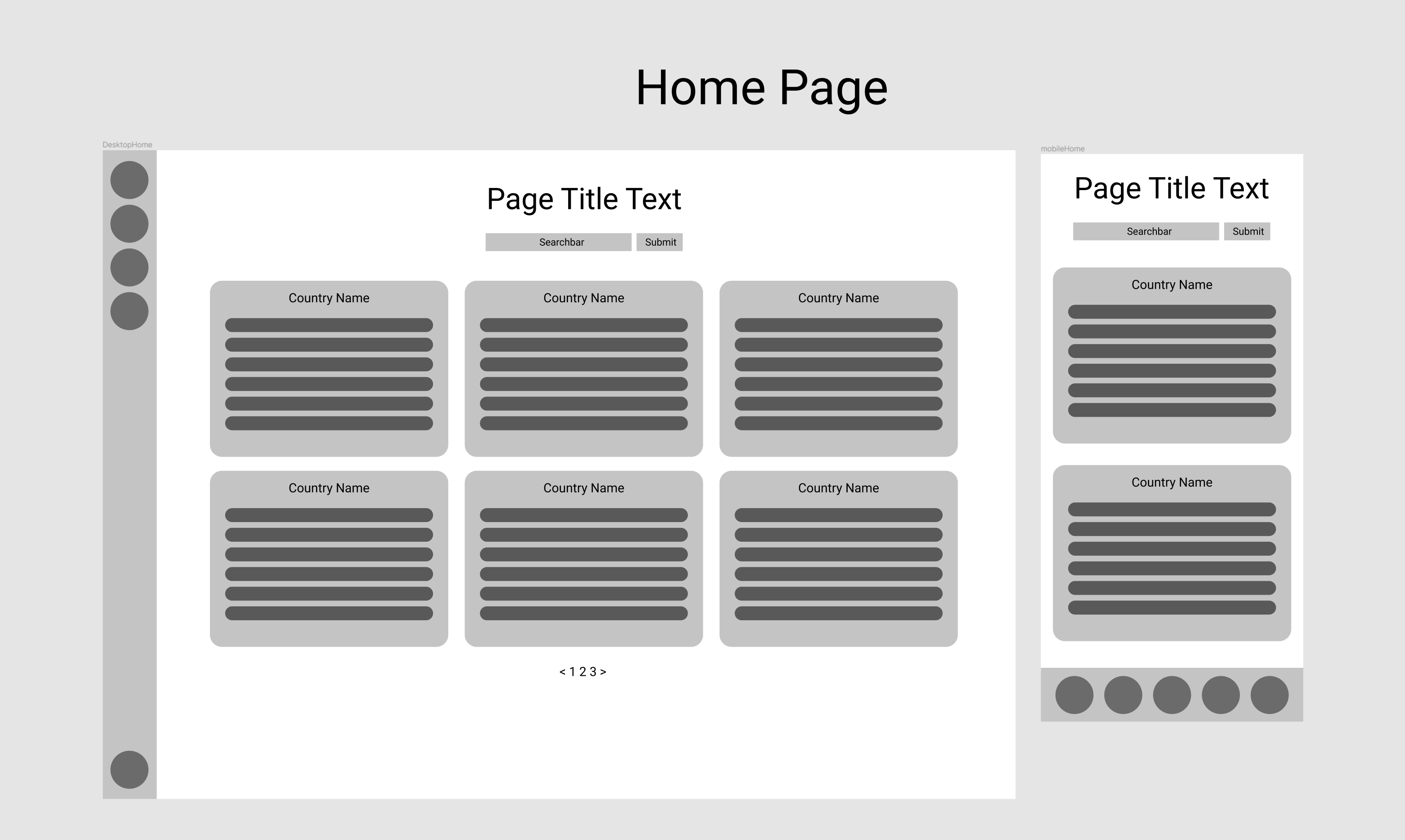
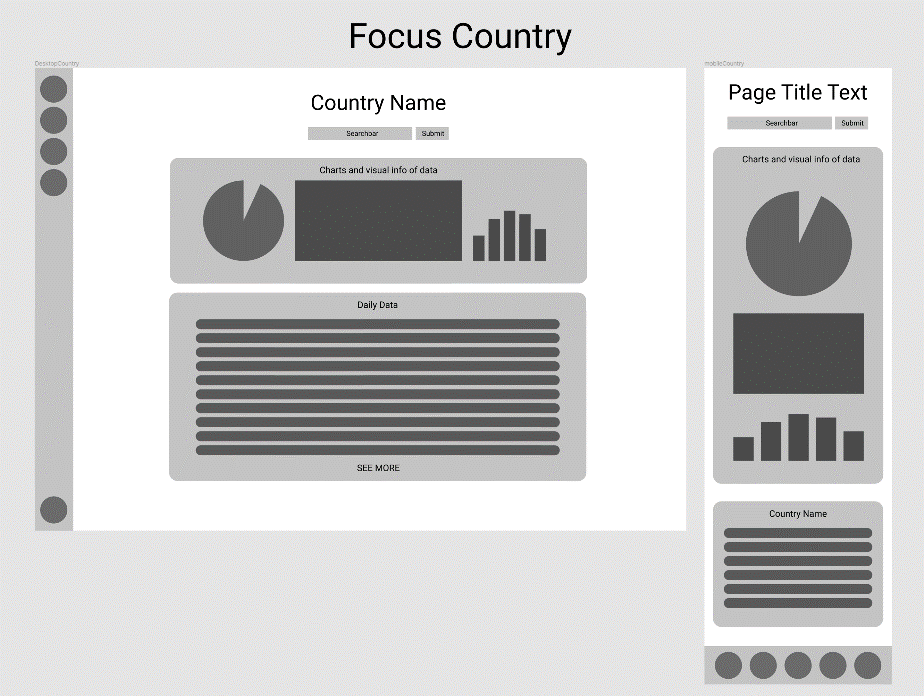
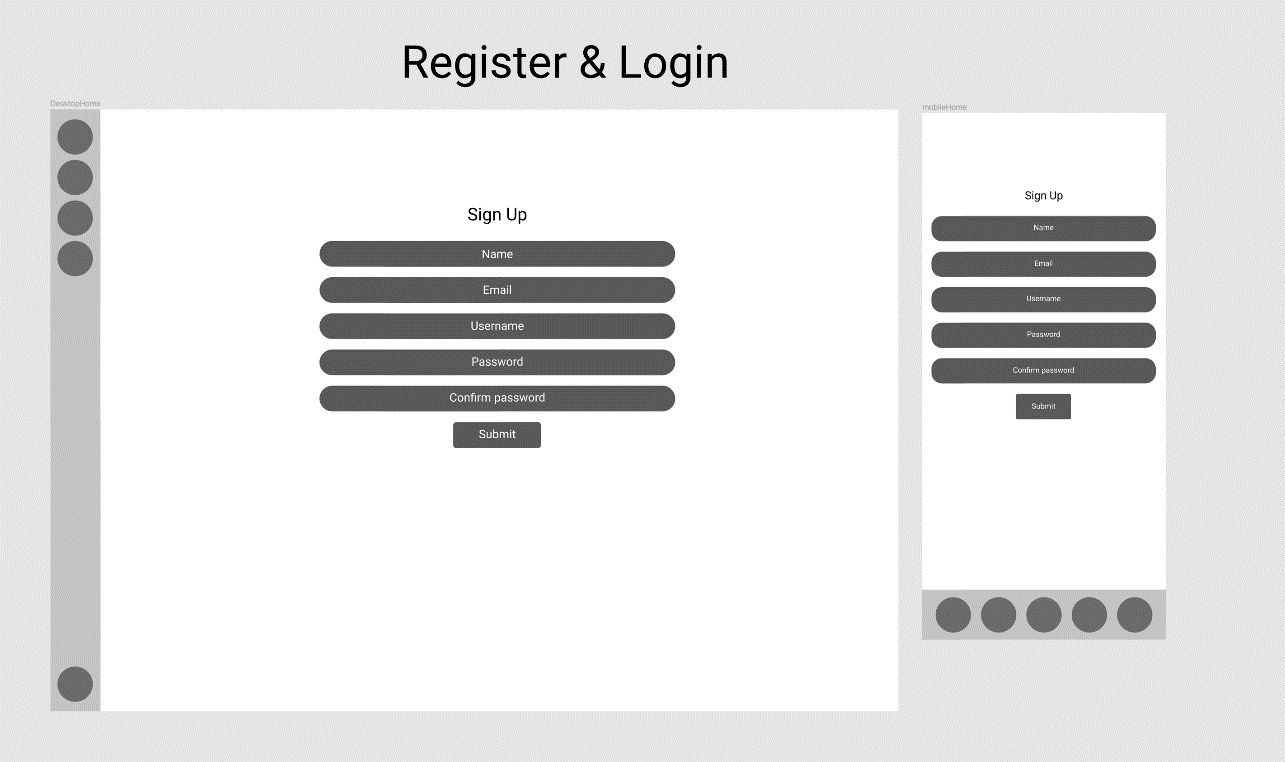
**Country search function.**I would like for a search function on the main page that will allow the user to search for the country they want to see the data for. The search function would be able to give auto finishing options depending on what letters have been entered in by the user, much like how Google tries to give suggested search options.

**Client account signup form.**I will have a signup form to allow users to create an account which will give access to private information or private pages inside the application. The intention is to showcase the ability to implement such a feature, however the actual functions given to the user may be partially arbitrary as the nature of a covid19 data aggregation don’t really require for a user account or admin system which was discussed with the Lecturer.

**Client login system with private access features.**this will support the Client account signup form in allowing the users to regain access to their account in the application after they have created an account with the sign-up system.

**Cloud hosting.**Cloud hosting is important to research and implement as this will ensure there is site reliability, and the service is always available to the users. Further research required as there maybe pricing issues that will prevent this service.

**Css styling and animations / Dark Mode.**I would like for good use of Css styling to ensure the data and UI is presented in a user-friendly manner with also a possible option for a dark-mode colour palette to help users with issues to bright light and to help prevent device battery wastage.

**DESIGN & SITE MAP**design inspiration:Wireframes:  
********

**What i learned:  
Ajax:**  
Ajax has allowed me to action database requests based and update them live to the DOM on event actions instead of reloading the entire page. This has been useful to reduce the amount of REST API requests and database queries that are being performed unnecessarily. And to show live search results to the users.  
Resource:<https://www.youtube.com/watch?v=tNKD0kfel6o>, <https://www.youtube.com/watch?v=XhMGV8PzyOg>  
  
**Prepared Statements:**  
a way to protect the data from users before sending the query to the database, preventing malicious attacks to the database.  
 **Regex:**  
I used regex notation to ensure the username is a correct format when creating an account.

**Postman:**  
I used Postman to .

**Hashed passwords:**

DATABASE UML:  
  
Website/files UML: