Balinder WALIA

Task 1: Steps performed:

Project: NodeJS v4.4.7 - Dashboard App to Analyse the UK Census 2011 Age structure data

Tools: MongoDB, Node.JS, plot.ly (D3 and react JS open source charting library), CSV to JSON npm convertor Module for Node.JS, Nginx High performance HTTP server.

Step 1.1: I download the Age structure data in CSV format

Step 1.2: Understood the format of the data and decided to create flexible approach to allow data in put into Dashboard in any format such as CSV, JSON and MongoDB
- https://task1.tenthmatrix.co.uk/data/KS102EWDATA.CSV (in CSV via URL)
- https://task1.tenthmatrix.co.uk/data/KS102EWDATA (in JSON via URL)
- Wrote a quick CSV to JSON converter generic functionality in NodeJS

Step 1.3: Dumped data into MongoDB and produced quick scripts to query it, produce it in JSON format scripts as it is easier to represent the columns and rows in Charts, and Visual UI elements from JSON

Step 1.4: I decided to work with HighCharts http://www.highcharts.com to start with as I have previously used it in projects and have a lot experience of it but while working I realised they have changed their licences scheme. So on Monday evening I switched to https://plot.ly

This change while in the middle of the task shows my **confidence** to get my hands dirty with even **new technology** without any **fear**. Also taken responsibility to shows that I am not prepared to compromise with quality even if it means I have to redo the work, I am not going to deliver something flawed.

Step 1.5: Implemented - Males versus Females (England & Wales and ability for users to drill down to sub areas and wards, Pie chart)

https://task1.tenthmatrix.co.uk/chart/chart.html

In this chart the data is picked up directly from CSV just to demonstrate the coding skills and flexibility. In other charts data is picked up by queries high performance MongoDB

Step 1.6: Implemented bar chart to show various age demographics data analysing the various age ranges in various regions of England and wales.

https://task1.tenthmatrix.co.uk/chart/chart.html

Step 1.7: Implemented an interactive bubble chart

https://task1.tenthmatrix.co.uk/test1

Step 1.8: Note: Implemented all charts and dashboard in mobile friendly jQuery bootstrap framework.

Step 1.9: Deployed this little Dashboard App on a Redhat linux box with MongoDB, Node.JS, plot.ly (D3 and React) with Automated Ansible Scripts. So that I make updates easily from my Mac to the server. The Deployment and Development Scripts take care of continuous deployment. It

shows the end to end development and deployment skills I have gained over the years through projects and or self study.

I can setup Ansible or Docker based auto scale deployment servers so that if the need arises due to high traffic on the servers we can easily scale it up and down. Also due to my expertise in NGINX means I can setup the load balancing and failover myself with full control over the code rather than paying a lot of money for the same tools and services to hosting providers and their consultants who are biased and have commission based money making schemes.

Input: CSV files

Output

Dashboard:

https://task1.tenthmatrix.co.uk/chart/chart.html

4 interactive charts:

https://task1.tenthmatrix.co.uk/test1 [Automatically increments the data using simulator to show how to update information in real time]
https://task1.tenthmatrix.co.uk/test4
https://task1.tenthmatrix.co.uk/test6
https://task1.tenthmatrix.co.uk/test7

Thanks