**Wealth Inequality Visualisation**

Contents

[Introduction and Problem Definition 2](#_Toc140672242)

[Research 3](#_Toc140672243)

[Source Data 3](#_Toc140672244)

[Who else is doing this? 4](#_Toc140672245)

[Stakeholders 5](#_Toc140672246)

[Problem Solution 6](#_Toc140672247)

[Overview 6](#_Toc140672248)

[Key Features 6](#_Toc140672249)

[Computational Methods 7](#_Toc140672250)

[Key Features 10](#_Toc140672251)

[Requirements 12](#_Toc140672252)

[Limitations 13](#_Toc140672253)

[Measuring success 14](#_Toc140672254)

# Introduction and Problem Definition

Poverty is a huge problem in the UK and around the rest of the world. This has risen in recent years with international events like Covid, Brexit and the war in Ukraine having a large impact on individual wealth. More recently the energy crisis, high levels of inflation and increasing mortgage rates have compounded the financial problems people are facing.

Wealthy people are often less impacted by such events and in many cases, with the help of professional advisors, can often profit from them. This wealth inequality has continued to grow over the years and shows no signs of slowing. One shocking statistic is that the richest 50 families in the UK hold more wealth than the poorest 33.5M families. That’s half the population of the UK.

A green rectangular sign with white text

Description automatically generated

Source: Analysis by Ben Tippet and Rafael Wildaur, University of Greenwich

The aim of this project is to help visualise the huge wealth of the super-rich and also show the impacts of redistributing this wealth. Often when we hear statistics involving large numbers it’s difficult to comprehend them. We know a billion pounds is a lot of money but it’s hard to appreciate its size. This project will help to contextualise these numbers and the dynamic visualisations we provide can be used by our stakeholders to help them redress the balance.

# Research

## Source Data

I’ve decided to concentrate my research in the UK. The Sunday Times produces its annual rich list which is published online and freely available:

A screenshot of a computer

Description automatically generated

Source: Sunday Times Rich List 2023

In order to calculate increases in wealth we’re using the Bank Of England’s Bank Rate which is currently 5% (<https://www.bankofengland.co.uk/monetary-policy/the-interest-rate-bank-rate>).

In contrast to this extreme wealth, poverty in the UK has been rising. According to Statista there were 14.4M people in poverty in 2022 of which 4M are children. Trussell Trust, the food bank, charity has shown large increases in demand over the last few years:

A graph of a graph showing the number of people forced to turn to a bank

Description automatically generated

Source: Museum of Homelessness

Definitions of poverty can vary and can be quite complicated. Also there are variations depending on region. For the purpose of our calculations, we will be using a figure of an additional £10k pa to move people out of poverty.

*Additional data sources also researched:*

* <https://equalitytrust.org.uk/>
* <https://www.jrf.org.uk/report/uk-poverty-2023>
* https://commonslibrary.parliament.uk/research-briefings/sn07096/
* <https://www.resolutionfoundation.org/app/uploads/2020/12/The-UKs-wealth-distribution.pdf>
* https://cpag.org.uk/child-poverty/child-poverty-facts-and-figures

## Who else is doing this?

I’ve researched other websites to see if any are producing the kind of visualisations I’m going to produce but can’t find any. I’ve used numerous search engines, e.g.:

* Bing
* Google
* Yahoo

Searching with many terms, here are a few examples:

* Wealth inequality visualisation
* Wealth distributions visualisation
* Interactive wealth redistribution
* Dynamic data visualisation

There are websites that allow you to contextualise extreme wealth:

A drawing of a cake

Description automatically generated

Source: NY Times

There are also lots of data visualisation websites but very few show dynamic content or allow interactivity and none of them do this with wealth as the subject. This justifies my decision to use this unique approach of having dynamic/interactive features to really help highlight and contextualise the issue.

## Stakeholders

As mentioned, the problem we are trying to solve is to help visualise the wealth of the super-rich and how redistributing this wealth could reduce poverty in the UK. This solution will be an invaluable tool for those working in this field. At a wider level, our stakeholders are anyone with an interest in this subject. This could be the general public, those in poverty, the super-rich, charities, large companies, foundations, media companies or government departments. However, it’s important that we interact with our stakeholders and obtain feedback at all stages of the project. For this reason, we’ve chosen to work directly with:

**Sam Clements, Finance Adviser for a commercial Bank**

Sam has worked for many years advising wealthy clients on the best way to invest their money.

**Marilyn Stewart, Charity worker**

Marilyn, now retired, still works closely with many charities and has a good understanding of the difficulties faced by those in poverty.

**Matthew Wright, Media worker**

Matt works closely with many media and marketing outlets and is familiar with many social media trends

I interviewed all of our stakeholders prior to the start of this project. I’ve summarised the feedback below and underlined the feedback that will impact the design. See Appendix: [Stakeholder Interviews](#_Stakeholder_Interviews) for a full list of questions.

### Interview Summary

**Sam:**

* [Positive](https://equalitytrust.org.uk/) about the idea
* All her clients are aware of poverty and will already make charitable contributions.
* Anything that highlights the amount or impact of poverty would be beneficial.
* Agreed that a visualisation that showed how changing the wealth of the rich impacts poverty could be affective. Being able to adjust this change in wealth would be useful (for example, if they were to give 1% of their wealth what would the impact be vs 5% or 10%).

**Marilyn:**

* [Positive](https://equalitytrust.org.uk/) about the idea but sceptical that there will be any impact
* Very aware of the struggles families are going through
* Very interested to see how the impact on wealth redistribution could impact poverty. Thinks this is a key part of the project.

**Matt:**

* Really likes the idea and thinks it could be big in social media if delivered effectively
* Thinks real-time wealth visuals could really raise awareness. Maximise impact by making it materialistic. How long until the interest on someone’s wealth would buy a phone or a car etc.
* Go for shock factor to gain traction in social media

## Computational Methods

Does this problem benefit from being solved by computational methods?

|  |  |  |
| --- | --- | --- |
| Method | Explanation | Justification |
| Thinking abstractly | Abstraction allows us to remove specific details and break down complex problems. | On the server side we will use abstraction to model the data we pull in from external sites. This will give us a high level view that removes unnecessary detail. On the client side we will use abstraction to simplify both the data analysis and also for writing our calculation engine. |
| Thinking ahead | By thinking ahead during both the analysis and design phases it allows us to build efficient, reusable code. It also helps avoid mistakes that could waste time. | The inputs to our system are well-defined:   * Wealth and poverty data for the server * User interaction for the client   The outputs are also well-defined:   * The server will output normalised data for the client and also serve the web pages. * The client will output visually interesting data   Data will be persisted to a local database which can be accessed if our data source is unreachable. It’s worth noting here that we won’t pull all data in automatically. For example, the interest rate is just one integer so can just be stored in our database or even in our calculation engine. There’s no justification here for using anything more complicated.  The data model will be used by the client and server side code to avoid unnecessary duplication. |
| Thinking procedurally | Thinking procedurally allows us to identify the individual components of a problem and its solution. We can then work out the necessary steps to build the solution. | The problem is the visualisation of extreme wealth and poverty and also the visualisation of the impact of redistributing the wealth.  The components for the server solution will be processed in this order:   * Sourcing wealth data * Sourcing poverty data * Normalising the data * Persisting the data * Delivering the data to the client   The components for the client solution are:   * Displaying the webpage * Requesting data from the server * Building a calculation engine to display real-time updates * Taking input from the user to input into our calculation engine and display the output. |
| Thinking logically | By thinking logically we will identify points in our solution where decisions need to be made and what conditions affect this decision. | By using this method it will enable us to deal with scenarios such as our data source being offline or if there’s a change to the source data format. This also helps us build in resilience to our solution. |
| Thinking concurrently | Concurrency allows us to tackle parts of the problem at the same time. | Our server side code runs independently from and concurrently with the client side code.  The client side calculations will run in a separate thread so as not to impact user experience. |

In conclusion, it’s clear that this problem is only solvable with a computational set of processes. Every part of this solution from data sourcing to calculation to display is more efficient using this approach. Also, given that the solution will deliver real-time, complex calculations then you could say it’s impossible to deliver it any other way. However, one example of where we

These methods have helped me produce the following architectural flow diagram:

## Key Features

# Requirements

* Stakeholder access to a web browser with javascript enabled
* Stakeholder PC capable of running a web browser doing real-time calculations
* Programmatical access to data sources
* Web hosting

# Limitations

* External data sources can be unreliable. For example, the source may be unavailable or the data format may change. Resilience will need to be built in to our solution.

# Measuring success

For this solution to be successful we will need to write up a set of use cases which can be tested and signed off. These will be written as part of the design stage and will help drive the development.

In addition, we will be asking for input from our stakeholders at the design stage and also will get their feedback after each delivery. This iterative, agile approach will ensure that the solution is meeting their requirements.

# Appendix

## Stakeholder Interviews