**BSC Computing Hons (Sandwich)**

Daniel Blake – 000888005

University of Greenwich

**An Investigation into the Use of Data Visualization Tools in .Net Web Applications for Analysing Data (UK Modern Slavery Act)**

**COMP1682 (G Wizards) Project**

# Abstract

# Acknowledgement

Contents

[Abstract 1](#_Toc5209895)

[Acknowledgement 1](#_Toc5209896)

[1 Overview 2](#_Toc5209897)

[1.1 Background 2](#_Toc5209898)

[1.2 Current Situation 3](#_Toc5209899)

[1.3 Keywords 3](#_Toc5209900)

[1.4 Aim 3](#_Toc5209901)

[1.5 Objectives 4](#_Toc5209902)

[2 Literature Review 5](#_Toc5209903)

[2.1 Introduction 5](#_Toc5209904)

[2.2 Data Visualisation 5](#_Toc5209905)

[2.3 Data collection 5](#_Toc5209906)

[2.4. Similar studies 6](#_Toc5209907)

[2.5 Conclusion 6](#_Toc5209908)

[Requirements 6](#_Toc5209909)

[Prioritisation 6](#_Toc5209910)

[Planning 7](#_Toc5209911)

[Frameworks and Methodologies 7](#_Toc5209912)

[Design 8](#_Toc5209913)

[Use Cases 8](#_Toc5209914)

[Class Diagrams 9](#_Toc5209915)

[Flow Diagram 9](#_Toc5209916)

[ERD Diagram 9](#_Toc5209917)

[Development 9](#_Toc5209918)

[Legal, Social, Ethical and Professional 9](#_Toc5209919)

[LSPEi Analysis 9](#_Toc5209920)

[GDPR 9](#_Toc5209921)

[Implementation 10](#_Toc5209922)

[Testing 10](#_Toc5209923)

[White Box 10](#_Toc5209924)

[Black Box 10](#_Toc5209925)

[Training 10](#_Toc5209926)

[Future Support 10](#_Toc5209927)

[Bibliography 10](#_Toc5209928)

[Appendix A: 11](#_Toc5209929)

[Gantt Chart: 11](#_Toc5209930)

# 1 Overview

## 1.1 Background

*“The Business, Human Rights and the Environment Research Group (BHRE) at the School of Law undertakes research on how companies and public authorities report under the UK Modern Slavery Act (****MSA****). The BHRE has to process several hundreds of reports and analyse them and needs an application which will help systematise the information of these reports, and produce meaningful outcomes which allow comparisons, including statistical data. This application should support BHRE researchers to point at trends in reporting and gaps that need to be addressed.*

*This will be developed as a web application. The ideal candidate must have good web development skills with proficiency in one or more of the following programming languages and libraries: C# .Net, Webforms, .Net MVC, HTML, CSS and JavaScript. They should also have good database design skills and must be familiar with Microsoft SQL server. Students who would like to work on this project must be willing to learn other .Net packages.”*

The School of Law at the University of Greenwich are looking for a web application to help them to process and analyse data to find gaps and discrepancies within MSA in both universities and local authorities for their reports. The client (Olga Martin-Ortega from the Law School), has stated that the they system they require needs to allow them to visualise data better than the Excel spreadsheets they are using currently. In the article ‘Using Web Application for Data Visualisation’, Dr Tony Glover explains that:

*“Modern society generates vast quantities of data; “big data” appears to be ubiquitous in present day terminology. The sources of this data are many and varied; financial, scientific, healthcare, economic, political, sports, architectural, social media; the list is seemingly endless. With all this data comes the need to visualise, interrogate and interpret all the information to extract some form of pattern or meaning.”* (Glover, 2016).

In the case of the system for the law school, they require the ability to generate and analyse large sets of data from different universities and public authorities to use within their published reports to back up their investigations into whether these companies are sticking to the modern slavery act. Using the various data visualisation tools available, it is possible to create a system that would take this data and provide it to the user in an easy to understand format for them to analyse.

This project will investigate the methods available for creating data visualisation within web applications, analyse them and find the one most suited for the system the Law School require.

## 1.2 Current Situation

The law school is currently using a variety of excel spreadsheets to collect data on various universities and local authority’s modern slavery act statements. This involves going onto every company’s website and reading over the entire statement and adding this into the excel sheets. Later, this data is then analysed using Excels tools and used within 2 yearly reports; one for Local authorities, one for Universities.

1.3 Keywords

**ASP.NET, Data Visualisation, Big Data, C#, Web Application, MVC, Kanban**

## 1.4 Aim

**To create a data visualisation web application for the law school to assist them in gathering and analysing data on MSA act statements from local authorities and universities/colleges**

## Objectives

**1 Research Report**

* 1. **Research the area of data visualisation in web applications**
     1. **Research 2 options for visualising data in web applications**
     2. **Research the use of MVC within .net web applications**

1. **Design Documentation**
   1. **Create Use Cases in UML**
      1. **Create Use Cases based on creating a report by a user with report creation abilities. This would be the same process for Admins**
      2. **Create Use Cases based on users view reports with report creation abilities and view-only abilities. This would be the same process for admins and report creation users**
      3. **Create Use Case showing how an Admin user can create users**
   2. **Create Class Diagrams in UML**
      1. **Create a full Class Diagram on how different classes interact with each other**
   3. **Create Sequence diagram**
      1. **Create sequence diagram on a user creating a report in the interface**
      2. **Create sequence diagram on a user creating a new user**
2. **Implementation**
   1. **Create the first throwaway prototype of the system that the Law school requires**
      1. **Create a few pages showcasing the input of data**
      2. **Create a visualisation of data input in graphs, with statements etc.**
      3. **Meet with the client, and get notes from the throwaway on how the system should look**
   2. **Create database system**
      1. **Create user database**
      2. **Create report storing database**
      3. **Create multiple databases for each tax year (i.e. 15-16, 16-17 etc.)**
   3. **Create a login system with create and view permissions for both**
      1. **Create page to log on to the system**
      2. **Create page to create users for the system, with access for create view users**
   4. **Create reports pages**
      1. **Create classes to create new tables every year to input data in**
      2. **Create reports pages for report generation. Include pages to search for these reports, and a class to create pages for these reports to be saved and edited**
      3. **Create graphs using the data visualisation I have chosen that will appear with the reports based on the inputted data. Include ‘tooltip’ like information on these pages**
      4. **Add the ability to save and edit these reports**
   5. **Testing the system**
      1. **Test log in system**
      2. **Test reporting to make sure everything appears as expected**
      3. **Test account creation**
      4. **Test for SQL injection**
      5. **Test to see if special characters break the system**
   6. **Create user guide**
      1. **Create a short user guide on how to use the system**
      2. **Create a short training guide for teaching people how to use it**
3. **Evaluation** 
   1. **Evaluate the best choice for data visualisation** 
      1. **evaluate relations between both data visualisation choices, and why they were chosen**
      2. **Evaluate what went well**
      3. **Evaluate what could go better**
      4. **Evaluate what went wrong**
      5. **Evaluate the next steps for the system (i.e. upgrades, functionality that wasn’t added etc.)**

**Time will be included in Gantt chart**

# 2 Literature Review

## 2.1 Introduction

In this project, the overall outcome is to create a system to allow the input and analysation of data based on the UK Modern Slavery Act. This data will then be displayed to the user in an easy to understand format using data Visualisation. Although data visualisation is the focus of this project, an investigation into similar studies and best practices within data collection will also be required to ensure that the input information is correct.

## 2.2 Data Visualisation

Data visualisation involves displaying a vast amount of data to the user using either a graph or a table. The overall aim of this project is to create a system to make it easier for the user to find the information they are looking for without having to look through a lot of data. For this, web visualisation will be used to display this information to the user. This is explained by Jiaxin Wu; who states that web visualisation not only lowers the cognitive load on the user when looking at data, but also enhances interactivity, allowing the user to hover over the graph and see a run down on the relevant data (Wu, 2010).

When creating a visualisation of data, it is important to make sure the data can be placed into a visualisation. This is usually achieved by ensuring the data that is being visualised can be measured. In the book ‘Graphing Statistics & Data’, the writers specify 2 different types of data to be used in graphs:

* Cross-Section data – 2 or more types of data that can be measured separate from each-other (i.e. height and weight)
* Time Series data – Data taken over a certain period (Wallgren, et al., 1996)

colours for the user

## 2.3 Data collection

## 2.4. Similar studies

## 2.5 Conclusion

# 3 Requirements

When developing a new system, it is important to gather requirements from the client to formulate the system. In the case of the proposed system for the Law School, the following requirements have been gathered from meetings with the client:

|  |  |  |  |
| --- | --- | --- | --- |
| Ability to input/edit university/ Local Authority data | Data from previous years must be stored | Assist with analysing data | Program must be written using MVC and .net tools |
| Upload a PDF containing the statement | SQL server database for storing data | Allow for users to view all data from all collected years | Ability to organise data |
| Ability to generate reports based on input data | Admin privileges for higher rules and creating accounts | Ability to log into the system | Ability to input/edit University/Local authority statement data |
|  |  |  |  |

From the above requirements, it is then possible to organise them into 2 categories; Functional and non-functional. This helps when discerning which parts of the system will be

|  |  |
| --- | --- |
| Functional | Non-Functional |
| Ability to input/edit university/local authority data | Data from previous years must be stored |
| Ability to input/edit University/Local authority statement data | Assist with analysing data |
| Upload a PDF containing the statement | Program must be written using MVC and .net tools |
| Allow for users to view all data from all collected years | SQL server database for storing data |
| Ability to organise data |  |
| Ability to generate reports based on input data |  |
| Admin privileges for creating accounts |  |
| Admin privileges for creating new categories |  |
| Ability to log into the system |  |
| Ability to add/edit categories |  |

## Prioritisation

MoSCoW analysis allows for IT analysts to prioritise functional requirements into 4 separate sections:

* **Must have:** Functionalities that are going to be implemented
* **Should have:** Functionalities that should be added to the system but aren’t necessarily required for the system at this time.
* **Could have:**
* **Would have:**

This in turn allows the analyst to make sure the most needed functionalities are added to the system, whilst other functionalities can be added later in the process, time permitting.

|  |  |  |  |
| --- | --- | --- | --- |
| Must have | Should have | Could have | Would have |
| Ability to input/edit university/Local authority Data | Ability to add/edit categories | Admin privileges for higher rules and creating accounts |  |
| Allow for users to view all data from all collected years |  |  |  |
| Ability to organise data |  |  |  |
| Ability to generate reports based on input data |  |  |  |
| Upload a PDF containing the statement |  |  |  |

# Planning

As stated above, this project will be using ‘Trello, which is SCRUM software used to keep track of sprints. This will allow for developers to keep track of the system development and set out a specific set of tasks for what should be done during creation of the system. Trello will work like a Gantt chart, containing the above objectives as tasks for development to be picked up and completed. Within these tasks in Trello, developers will be able to comment on how far they have gotten, what has been implemented and how far it has gotten. It is possible to set the tickets as ‘In review’, for them to be checked and made sure they won’t break the system. From here, any comments on the tasks can then be used to form documentation on how the system was created, used, and how to fix any issues that may arise.

## Frameworks and Methodologies

This program will be created using ASP.net/C# using MVC (otherwise known as Model View Controller):

ASP.Net allows the creation of the system the client requires in the form of a web application, allowing it to be viewed on multiple different devices without installing a program on the user’s device. Dr Tony Glover (2016) has said *“As the visualisation is running inside a web browser then, in theory, they can be executed on any web-enabled device. This means that they may be viewed on a whole range of devices in a whole range of differing scenarios.”* (Glover, 2016). This means that through the web application, it is possible for the client to access the program on multiple devices almost instantaneously, as well as provide a URL for the graphs and data for the client to reference within their reports. As well as this, data visualisation tools will be used to display data in an easy identifiable way, without losing the visualisation that has been created when the client moves device.

In the journal ‘A Novel Web Application Frame Developed by MVC ‘, the authors state: *“MVC consists of three kinds of objects. 1) the Model: is the application object, 2) the View: is its screen presentation, 3) the Controller: defines the way the user interface reacts to user input.”* (GuangChun, et al., 2003)*.* This means that it is possible to re-use models and objects with various views instead of re-writing code and making the program take up more space of the server. This will also assist with future maintenance, as the use of these objects for repeated code allows most changes to be made in one place. This is the same for upgrades to the system, as they would only need to be added into the relevant model/view for the controller to use.

An example of a way MVC would be used in the proposed system is when generating graphs:

* A **Controller** to take information on how the graph should look, what graph etc.
* A **View** to take in information from the controller about the graph and generate the graph with data in it
* A **Model** to take the information from the view, to pass back the relevant information on the graph it requires, colours, size etc.

As this system has a small period for development to be completed, it is best that the RAD frameworks are used to ensure that the product is developed quickly, efficiently, and at a low cost. Because of this, the best methodology for this project would be SCRUM within the software ‘Trello, as this means the project can be separated into manageable ‘Sprints’ with a part of the system complete at the end of these sprints.

Trello is a SCRUM Software like Jira, which allows for the developers to keep track of the development cycle using ‘user stories’, which are used to describe work items in a non-technical language, and the outcome of that task by the end of a sprint (i.e. ‘As a user, I should be able to log in’). From here, these stories can be added into a backlog, prioritised, given an end date of when the task is complete and when the sprint will end. This should allow the developers of this software to keep track of tasks, know what the next steps are and keep to a clear deadline of each sprint.

## Design

When designing the system, UML will be used to create use cases, class diagrams and sequence diagrams to lay out how the system will work. For example; Use Case’s will be used to create diagrams on how users will create reports, Class diagrams can be used to show how different classes and views will work together within the program, and a flow diagram to show how the user will navigate and use the system.

### Use Cases

Within the proposed system, use-cases have been created to show developers how different users and sub-systems interact with different functionalities.

### Class Diagrams

### Flow Diagram

A flow diagram is used to show how the user and data goes through the system from start to finish.

### ERD Diagram

The ERD (Entity Relationship Diagram) assists database engineers when planning how data should be stored within the proposed system.

## Development

During development, a basic throwaway prototype of the system will be created to show the client how the system will look, and how their reports will be displayed on screen. From here, it will be possible to find out any extra requirements and specifications that the client requires. After the first throwaway prototype, development work will start on evolutionary prototypes, with regular meetings with the client to make sure everything is up to their expectations.

As this project is going to be written in ASP.net, a computer installed with Visual Studio and a version of MySQL server will be required to create and test the proposed system. The computer in question is also required to be powerful enough to run both the database server and host the website at the same time. Copies of Microsoft Visio will also be helpful for the creation of UML diagrams. A Git server will be used when saving the new iterations of the system, making sure that should any issues arise with any computers that the program is still saved elsewhere. This also helps with version control, meaning that if an issue occurs during development there is still a rollback point.

# Legal, Social, Ethical and Professional

## LSPEi Analysis

The MSA data the proposed system will be holding is available in the public domain and is accessible for anyone to access, therefore a login for viewing data isn’t necessary. However, as this data is for the universities use a login will be implemented to stop anyone who shouldn’t be on the system from manipulating and viewing the data collected. This is because the information in question is sensitive to each company it is taken from, which could mean that a ‘corporate espionage’ like attack could be launched on the system to remove data and ruin the validity of the report’s created.

## GDPR

The system won’t be collecting or recording user information so GDPR won’t be applicable, but user information (i.e. full name, emails, passwords) will be stored within the database. This means that precautions will need to be in place to protect the client’s data such as password encryption, permissions for specific users, and measures will be put in place to stop SQL injection occurring. Verification methods will also be set up to ensure the right emails are sent the right sign up codes, and the right users are able to access the right areas. Fictitious names and passwords will be used during this project’s creation, so the hiding of email address and names won’t be required at this stage.

# Implementation

## Testing

### White Box

### Black Box

## Training

## Future Support

# Bibliography

Addepar, I., 2018. *Ember Charts Documentation.* [Online]   
Available at: https://opensource.addepar.com/ember-charts/#/overview  
[Accessed 30 10 2018].

Artigas, F. & Chun, S. A., 2013. Visual Analytics for Open Goverment Data. *Proceedings of the 14th Annual International Conference on Digital Government Research,* pp. 298-299.

Atlassian, 2018. *How to do Scrum with Jira Software.* [Online]   
Available at: https://www.atlassian.com/agile/tutorials/how-to-do-scrum-with-jira-software  
[Accessed 2018 10 30].

Bostock, M., 2017. *d3js Documentation.* [Online]   
Available at: https://d3js.org  
[Accessed 30 10 2018].

Glover, D. T., 2016. Using Web Applications for Data Visualisation. *Proceedings of the European Conferene on Cognitive Ergonomics,* pp. Volume EECE ’16, article 31.

GuangChun, L., Lu, W. & Hanhong, X., 2003. A Novel Web Application Frame Developed by MVC. *ACM SIGSOFT Software Engineering Notes,* 28(2), p. 7.

Microsoft, 2018. *ASP.NET MVC 5 Documentation.* [Online]   
Available at: https://docs.microsoft.com/en-us/aspnet/mvc/overview/getting-started/introduction/

Rodríguez, A., 2018. *LiveCharts Tutorials and Documentation.* [Online]   
Available at: https://lvcharts.net/App/examples/wpf/start  
[Accessed 2018 10 30].

Wallgren, A. et al., 1996. *Graphing Statistics & Data.* 1 ed. Halmstad: SAGE Publications.

Wu, J., 2010. Web Data Visualization Based on the Function and Implementaion Dimension. *2010 International Conference on Data Storage and Data Engineering,* pp. 245-249.

# Appendix A:

## Gantt Chart:















