CS958 Project

Coursework Assignment

Lewis W. Britton
202194412
University of Strathclyde
Glasgow City, Scotland

Burning Roots:

Written & Directed By
John Hughes
Executive Producer
Michael Mann
Created By
Anthony Yerkovich
Music Composed & Performed By
Jan Hammer

... Words

Dissertation Submitted in Partial Fulfilment of the Requirements for the Degree of Master of Science Software Development at the University of Strathclyde

Academic Year 2020/2021

Abstract

Index terms:

Declaration & Information

This dissertation is submitted in partial fulfilment of the requirements for the degree of Master of Science Software Development at the University of Strathclyde. It accords with the University's regulations for the programme as detailed in the University Calendar.

Mail To: wi.lbritton@yahoo.com	Telephone: 07415 212 ***
Website: lewisbritton.com	GitHub: FedeRog1977

This document's presentation reflects the use of LaTeX typesetting (Figure B7), using Computer Modern Unicode (Figure B8) (I haven't reached my GNU Troff phase yet). This escapes the inane formatting requirements of my institution. References are presented using BibTeX, favouring *oblique* over *italic*, in-line with Donald E. Knuth's preference (Knuth, 2020). The process is executed in command line using Vim, which is a very powerful editor that has many commands, too many to explain in a tutor such as this. For maximum optical pleasure, the use of MμPDF is vigorously advised with [-I]. Navigate this document using h(←), j(↓), k(↑), l(→), ensuring that the Caps-Lock, Super-Key 'mod', or any other command key is not depressed. Note that the Oxford Serial Comma is favoured throughout this text. This study's sentence structure focuses on pragmatics and syntax, disregarding bloated filler content. Arguments are coherent, logical, definitive and straight-to-the-point. Nugatory theory is ignored. If you are curious about any of the mathematical, operational, logical, etc., symbols or notation used in this report, a comprehensive LaTeX-syntax-based symbolist will be available from my website library from approximately summer 2021.

The word count of this piece reflects relevant content from titles, heading classes 1, 2 and 3, paragraphs, footnotes, tables (excluding [results] tables 4.7, 5.10 → 5.20), table titles, figures, and figure titles in *Chapters 1 → 6.* Word count excludes any pre/succeeding content from *Abstract*, *Declaration & Information*, *Acknowledgements*, *Table of Contents*, *Appendices*, and *Bibliography.*

I declare that this document embodies the results of my own work and that it has been composed by myself. Following normal academic conventions, I have made due acknowledgement of the work of others.

Signed:

Date:

Acknowledgements

I would like to thank my dissertation supervisor, Dr Devraj Basu, for his approach with regards belief that students must be self-disciplined, organised, structured and punctual to their own degree. This closely relates to my own personally practiced work ethic and philosophy of the ASAP standard.

I would like to give credit for the computational aspect of this study to one of my biggest inspirations, John "The Tzar" Kelly. He inspired my love for everything bare-bones computational, from simple arrays (of hope), through Hyperthreading-enabled, all the way to x86 Assembly. I would also like to accredit Luke Smith for the foundation of my knowledge of Bram Moolenaar's Vim and LaTeX. This study's presentation would not have optimal without Smith (2015).

This piece would not have been as efficient without the aid of the only acceptable Linux distribution, 'distro' if you will, Arch Linux. I would like to thank Judd Vinet for his eye-opening and life-altering contribution to the development and computer-system enthusiast community. 'The Arch Principal' is certainly out in high force. Finally, for making use of this software mechanically efficient, I would like to thank IBM for the creation of the ThinkPad T23, X30, T42, R50e, T60, X60, X200, X220 and T420 neoVimPads, the UltraDock, and the 1987 Model M *Catastrophically Buckling Compression Column Switch and Actuator* typehorse (US369 9296A, 1972). For your convenience, one of my blog posts can satisfy your interest in this.

Table of Contents

List of Tables

List of Figures

1 Introduction

1.1 Purpose & Industry

The system developed throughout this project is a functional web application based on providing user-location-based and external GPS data. User-based GPS services are based on data relevant to the user's machine and external services are based on context-relevant data. The system, which so forth may be referred to as 'the system', 'the application', 'the site', provides detailed information, guidance and recommendations relevant to sport and leisure, particularly hiking and its associated practices, in Scotland's fine rural outdoors and [*these mist covered mountains, which are home now for me*]. Therefore, upon a hypothetical release of a full version of this system, it would be a direct competitor of services such as Walkhighlands, Strava, Garmin Connect, AllTrials, etc. Due to the autistic and comprehensive nature of its development, it would be in the market not in the competition-driven business, but in the [empire business].

1.2 System Structure

The site upon which the system is spread is static, not dynamic, meaning any possible requests made buy the user are based on existing data. The site does not reference an external database for any purpose. Thus, no PHP or SQL-relevant content. The system is segmented into four, approximately equivalent, parts with the metric(s) determining their weight being algorithmic volume, number of services, etc.

The first, 'home'/'drafting room', page allows the user to view a comprehensive overview of the site's offerings. It allows the user to quick-view activities in the 'overview' section; analyse their projected personal ability and their projected gear performance (upon their input(s)) in the 'conditioning' section under 'ability' and 'equipment cache'; and, view a coordinate-based weather briefing in the 'weather' section. All of these sections include 'key's and 'suggested reading's. Yes, this site is that obnoxious.

The second, 'conquest map', page allows the user to view and interact with various GPS and mapping features including their location and location seeking. It also includes functionality which allows pinpointing of particular features upon interaction, such as Munros, Munro Tops, Corbetts, Corbett Tops, etc. Further it includes aspects which allow the user to seek, be recommended, select and print GPX routes on the map.

The third segment is the 'ranger calculator' which is used purely for analytical purposes and allows the user to input data relevant to their ability, equipment, routes, etc. and will deliver output in the form of statistics tables and charts based on various computations. This element does not implement any GPS functionality in-line with the system's primary focus however, is extremely relevant.

The final segment is the 'general search' function which allows the user to input or select search criteria which returns a comprehensive overview of all statistics relevant to the match(es). Unlike the other sections, this is more subjective and informative, as opposed to being logical / statistics-oriented. That is, is exists more so for the users understanding of what they're doing and how to actually interpret some of the statistics they're being delivered in other elements. It is open to their use and interpretation.

The service is called '*Burning Roots*'. No, the rhetorical use of satiric misspelling is not unintentional malapropism; it is in fact deliberate. In harmony with the feeling you'd experience when [racing south-west to Lone Stallion Ranch], the term references that certain '*burning* desire' for freedom in the sweet country air, the one you only experience when digging to your deepest '*roots*' to achieve a new personal record or firmly assert your dominance over your inferiors. This service uses and computes data to encourage a user to take to the trails with motivation to be the fastest, most efficient, most prepared and most endured athlete on their *routes*.

1.3 Users & Platforms

All users of this system will be sport/leisure oriented and as this is focused on a specific group of enthusiasts who have a firm set of beliefs and a strong pre-developed relationship with their sport (lifestyle), it will likely only receive traffic from athletes who are already hiking-inclined. It may encourage new hikers due to the comprehensive nature and customization opportunity of the learning and planning material however, it is unlikely. The most efficient empires dominate only one type of market. This market does however expand to: walkers, hillwalking enthusiasts, scrambling-inclined 4x4s, [T6 vanlife] climbers, and Scottish mountaineers / ice climbers.

Upon original briefing, this system was planned to include a road cycling section which would essentially mirror the hiking part with cycling-specific data. However, upon reflection this element is irrelevant for two reasons. The first being all functional areas are covered by the computational processes involved with hiking data. And second, road cycling has little association with hiking and therefore the adjacent sport may be seen as irrelevant by a specific user. If it were to be included, it would only be logical to implement a wider array of sports for example, excluding road cycling and including mountain biking and (fell) running, which are actually relevant to hiking. Or an even wider array if road cycling were to blend in seamlessly. This is unnecessarily [time-consooming] and only duplicates processes and would not deliver additional benefit, only diminishing returns, upon the project marking process. I'm not [the Zuck'], not only do I not have the time or resources for this, I do not have the relevant background knowledge.

As far as platforms go, the user arrives at this site through a web browser. This system is deployed as a website and is therefore extremely versatile and usable on any device. Browser caching of script elements allows a user to view and interact with the relevant data offline, provided they receive GPS signal. For example, they can still view their location and route on the 'conquest map'. Of course, [this means that] the site is constructed using HTML, CSS and JavaScript. Elements of JavaScript allow this site to dynamically scale to various device sizes tailor relevant content to these devices.

1.4 Development Process

As this is a [solo project], it's one man, his [ThinkPad X220], [Artix Linux] and his [neoVim] setup. There is little requirement for extensive use of any formal [inane] project management methods such as team-based allocations or associated time-based or progress management coordination frameworks. Therefore, any adherence to processes aimed at mapping management of this project are / have been more logic-oriented and variable, allowing creative freedom. I work on an ASAP basis so one creative day of thinking may be followed by a [5am – 11pm] of implementation, which may then be followed by 2 days of idling. Any formal micro-level plan would be redundant.

Succeeding acquisition of user requirements, the most important part of gaining an understanding of how this system would look and operate is determining how the user interacts with the various aspects of their sport. That is, what data/inputs must the user provide, how will this be used, and what will it be used for to satisfy the requirements. Therefore, the mapping of the functional structure of the user interface (UI) leads this in the sense that it demonstrates the logic and process of interaction relevant to this data. Thus, this creates a valid starting point. And so forth, the development process of this project reflects what follows:

Requirement analysis
→ Structural design
→ Usable data construction
→ Graphical user interface functionality
→ Usable data implementation
→ Graphical user interface graphic design
→ Testing

Following the structural mapping, there is little sense in proceeding without any data to work with as incremental testing of site functionality would be challenging to impossible. So, it's at this point which the acquisition of relevant data takes place. In this case, this data accounts for non-user-centered data such as GPS coordinates, map regions, landmasses and their attributes, etc., which are essential for the majority of computations. Therefore, not only is [JavaBloat] implemented to manage site dynamics, it also computes based on data from these discussed files, in JSON format. It is only after this when the graphic design of the site can be allocated more focus, however of course much of it comes instinctively along the way also. After this, and frequent incremental developer tests, the system is ready for more expansive developer and user testing.

For the natural ease in workflow, for the developer's mental state, and for the minimization of [nugatory] methodologies and [bloated] task flow [cargo donkeys] such as IDEs and [froymeworks], all files (including 'code', data files, notes and write-up) are [composed and performed by Lewis Britton] in [Bram Moolenaar's Neo Vi Improved], in the command line of a pragmatic dwm setup on Artix. HTML, CSS and JavaScript is written from scratch in plain text format, therefore using no environment prompts or assistance, in order to keep the process practical. All write-up documentation is transcribed using [Donald 'Don' E. Knuth's TEX]. Or as some [neomoderinists] like to use, LATEX. Due to time constraint, there is unfortunately no mastering the fine art of [GNU Troff], so transcriptions may not appear [optically optimal] without the famous Groff-PostScript [multi-kill].

1.5 Disposition

So forth, the following elements of this project are responsible for...

2 Research & Evidential Background

Explores the areas of research and data gathering including hiking routes, hiking equipment, personal fitness and ability, geography and geology. Furthermore, presenting and examining results and conclusions to evaluations of currently existing competitors' services. This section acts as a literature review would in a paper based on, say, an empirical piece investigation; providing the foundational material upon which development aims to further succeed and 'develop'.

3 Data Processing & Methodology

Mapping and justifying the data selected for use in the system. This is broken into three segments, first being data acquisition which explains how and why data is sources from third parties and inputted from users. The second section explains how this data is manipulated and the third; statistical and informative output.

4 System Requirements

Describing the scale and scope of the users and their requirements for this system, and mapping how these are prioritized at the beginning of and throughout the project. This is in the context of functional and non-functional requirements.

5 System Design

Displaying the structure of the system architecture and how the logic aligns with the requirements of the system. Also, describing the various aspects of the user interface's functional and graphical communication and design process. It's apparent at this point how the data structure is made relevant to the design of the system using the requirements.

6 System Construction & Implementation

Providing a closer look at and justification of the development environment, languages and protocols selected for the creation of this system and exploring the various APIs and JavaScript libraries and other supporting tools used to enhance the system and allow it to function in harmony. Also, Providing an overview of how these elements were implemented from a project management point of view.

7 Evaluation

An evaluation of requirements gathering and the feasibility and tangibility of their implementation, an review of self-testing methods and additional tests, demonstrations of prototyping and various other aspects of developer' and user-centered testing.

8 Development Conclusions

Summaries of development conclusions which are presented pragmatically as objective, critical notes and possible segues.

2 Research & Evidential Background

2.1 Areas of Exploration

This system is designed to combine and present aspects of the different types hiking routes and their attributes relative to their conditions; the recommended and available equipment for users to investigate and explore expansive opportunities within; the personal fitness and ability level of users and therefore, their ability to interact with different routes and opportunities; and, the geography and geology of various aspects of hiking routes which contributes to various other factors within user ability how users may interact with the routes themselves.

2.1.1 Hiking Routes

The hiking routes are the foundation of this system. They are what gives purpose and reason to the GPS aspect of the system. There are various demonstrations of how hiking routes are implemented in different ways across slightly different platforms. Walkhighlands for example, presents very static use of these; displaying a page per routes listing manually expressed data and literature. Each route cannot be interacted with and had no dynamic attributes. They are simply listed for user interpretation.

2.1.2 Hiking Equipment

2.1.3 Personal Fitness & Ability

2.1.4 Geography

2.2 Material Investigation & Heuristic Evaluations

2.2.1 Walkhighlands

2.2.2 Strava

2.2.3 Garmin Connect

3 Data Processing & Methodology

3.1 Data Acquisition

3.1.1 Location

3.1.2 Self-Declared Attributes – Ability

Body statistics: height, body mass, body fat, muscle mass

Basics: average resting rate, 5k run average rate, 30 mile ride average rate, 10 mile hike average rate, 20 mile hike average rate

3.1.3 Self-Declared Attributes – Equipment

3.1.4 Weather

3.1.5 Route Data

Overview no longer gives an overview of user stats on routes, it now breaks down the routes.

3.1.6 Regional Data

3.1.7 Landmass Data

3.2 Data Manipulation & Output

3.2.1 Route Overview

Basics: distance, elevation gain, estimated time (based on ability, equipment), estimates energy output (based on user stats, ability)

Difficulty rating: playground, normie, enthusiast, trad, gigachad

CONQUER ROUTE

3.2.2 Weather System

3.2.3 'Ranger' Calculator

Route recommendations and suggestions

Risk indicators: lack of visibility (based on weather), misdirection (based on weather and terrain), falling off cliffs and ridges (based on weather and terrain), rock fall (based on terrain), rock kick-back (based on terrain), dehydration (based on weather and duration etc.), sunburn (based on weather and duration etc.), confrontation with livestock etc. (based on terrain and terrain type), plants and allergens like moss and ferns etc. (based on terrain and terrain type)

Winter risk indicators: lack of visibility - white-out (based on weather and terrain), cornices (based on weather and terrain), avalanche (based on weather), ice fall (based on weather), hypothermia (based on weather and ability)

Recommended gear: based on all route factors

Recommended climbs (based on terrain and equipment): trad, sport, top rope, bouldering, free tool, free solo (suggested reading table)

Ranger graphs: elevation profile, speed input, power input, heart rate input; Select: Constant Speed (Max); Output: Required Time and Power; Select: Constant Speed (Average); Output: Required Time and Power; Select: Constant Power Output; Output: Required Speed and Time; Input: Target Speed; Output: Required Time and Power; Input: Target Power Output; Output: Required Time and Power; Input: Target Energy Output; Output: Required Time and Power

3.2.4 Regional & Landmass General

3.3 Information Acquisition & Output

3.3.1 Purpose

3.3.2 Keys & Suggested Reading

4 System Requirements

System requirements gathering is essentially the stage at which the concept of a system's functionality is aligned with real-world user desires (requirements). This stage provides a context for creativity and a blueprint upon which this can be mapped. Although the founder of a system or concept may have a clear vision of the intended outcome of their development, understanding what final users need and want allows the creator to constantly tailor development. This may extend to how they and/or the software could/should gather information, store data, transact data, and output data/information.

4.1 Scale & Scope

I repeat, this is a [solo project] so there are some restrictions regarding the overall scalability of the project as a whole. If this system were to be designed by a team of professionals, it would be very large-scale and offer much expansible functionality. However, due to the number of personnel assigned to the task (a.k.a. me), the time constraint, and budget constraint, the system finds itself with two major general downscales: [1] there is no user-data back-end, meaning user accounts are unavailable on this system, which is acceptable as the static functionality of the site is most relevant; and, [2] the sample region for data collection is significantly smaller than that offered by other services, which is also acceptable as adding a wider scope of data (to the master JSON in this case) would only consist of repeating the same patterns perpetually. As this system does not aggregate this data for any form of cross-sectional statistical analysis, larger sample sizes become irrelevant after a certain point. Overall, this implies these factors are not directly related to any implemented requirements. Many of the specifics of these restrictions, and others, are discussed latterly in the *Evaluation* section.

In an ideal world, users would be able to expand their scope of interactions within the system (i.e. different sports with a wider and varying array of attributes), and the individual scale of these. [This means that] as sports differ, attributes and statistics differ, and information and guidance differs; offering a better-rounded service. Ideally, users should have the opportunity to fully customize their experience however, to do this on such a scale feasible with the development team available (myself) would be a significant over-effort for an under-achievement. This is why user accounts have been disregarded. This decision helps keep the system and implementation of requirements more manageable and makes it easier to achieve a polished product within the time constraint.

4.2 Gathering & Prioritization Methodology

In this scenario, there are three methods of requirements gathering and inference. The first is the purely user-centred method. As I am surrounded by people who share an interest in the form of this system, including professional developers, student developers and various other [NPCs] of the sort, they act as an accurate representation of market users as they share the same attributes. The second mode is alike however, is argued to be subject to various aspects of contextual bias. This consists of creative direction explicitly from the creator, me. In this context, these inputs will generally align with those of the formerly discussed however, due to the bias, is not considered a viable user-centred method unless used in conjunction with others affirmative of said criteria. The final method involves basic inference from the *Research & Evidential Background* section. That is, much of the material investigation of Walkhighlands, Strava and Garmin Connect in this section highlights areas for linear development. These refer to aspects which generally do not require user-centered input and must be developed purely functionally.

Many aspects of this system are purely functional and exist to serve an objective purpose. For example, relaying GPX and JSON data related to components such as route and weather information. [This means that] the primary functionality of the system is majorly accounted for in the latter of the three requirements gathering methods, in that the goal of the system is to create wider-scoped versions of much of the existing content in the explored areas. Therefore, the two former methods generally account for improvements which can be implemented upon these predecessors throughout development, which are primarily focussed on enhancing the user's experience. Myself and the discussed group of experienced others are a credible source for this, considering the scale of the project.

Randomly assigning requirements to development would be irresponsible. To help better-address the importance of the components of the required functionality of a system, requirements should be analysed using some form of hierarchical tool which highlights a clearer path for the development process. In this case, the Must-Have/Should-Have/Could-Have/Won't-Have (MoSCoW) methodology is selected for this purpose.

This helps differentiate between what functionality is essential to make the system run as intended, what functionality is required for optimization of the system, and what is required for additional enhancements. This approach generally shows functional requirements to average at the top-priority end and non-functional requirements to be distributed further down the hierarchy. With regards to the framework itself, note that:

- *Must-Have* implementations refer to aspects of the system which must be present in order to make it basically functional and behave as it is intended and as the user desires;
- *Should-Have* implementations refer to features of the system which should be implemented in order to make the essential features of the system more accessible and useable to the masses. They may also exist to improve efficiency, but [fly Under the Radar] and therefore go unnoticed by the user. They may offer additional functionality which makes the system more unique (or something of the sort) and therefore, more 'creative' and attractive to users.
- *Could-Have* implementations refer to aspects which may become present or relevant during the development of the former two. They may add additional functionality or usability to existing aspects or simply add final touches to the system overall. They are sometimes more contemporary.
- *Won't Have* non-implementations refer to aspects of the system which aren't necessarily impossible or are of a nature which the system 'can't have' but, which will probably be omitted or postponed due to constraints such as time, manpower, technical ability, finance, etc.

There is no formal client base for this system which means that no face-to-face client-oriented interviews or surveys can take place with regards to determining specific user desires. However, the aforementioned three-method protocol leads to the subsequently discussed requirements. To reiterate, the following list of requirements is generated through discussion of the desires of [1] an existing group of users of systems alike (some of whom are developers), [2] myself, another existing user of the sort, and [3] analysis of other services. This list is of course not exhaustive, as there is always room for perpetual development. However, it does account for every currently visible desire and possibility given the constraints.

4.3 Requirements

As discussed, it's important at this stage to clearly differentiate between essential functionality for the foundations of the system, generally accounted for in *functional requirements*; and functionality which more contemporary, generally accounted for in *non-functional* requirements. This is often considered 'making things *work* and making things *relevant*'. It is also a useful method from which to infer associations between elements of the design and construction stages of development.Retaining these requirements, their position, and hierarchy at the centre of development throughout the process allows accurate amendment of software and/or requirements along the way as more functionalities and possibilities become apparent and tangible or alternatively, further from reach.

4.3.1 Functional

4.3.2 Non-Functional

4.3.3 User Stories

Table 4.1: System Requirements & User Stories

4.4 User Accessibility

This system is available as a website application, it could be accessed and used by anyone with access to the internet. Therefore, it is to some degree essential that users of all types are welcomed to the site, even if they have no prior knowledge of the industry or field. More importantly, the graphic design and methods of interfacing in the user interface should be accommodating of users with possible imparities, such as a [severe mental imparity], by taking a user-centered approach which considers visible accessibility controls which allow these users to interface with the system more easily. Accessibility does not however only refer to disability. Ensuring optimal accessibility also accounts for factors such as implementing clean and concise CSS and graphic design for simple and intuitive navigation, stimulation, relevance, etc. These approaches are widely considered and implemented.

5 System Design

5.1 System Architecture

5.2 User Interface

5.2.1 Logical Design

5.2.2 Graphic Design & Communication

5.2.3 Interface Tree

5.2.4 Use Case Diagram

5.2.5 Primary Use Case Examples

5.3 Data Structure

6 System Construction & Implementation

6.1 Development & Languages

6.1.1 Development Environment

6.1.2 Front-End & User Interface

6.1.3 Semi-Rear-End & Supporting Data

6.2 Application Programming Interfaces (APIs) & Libraries

6.2.1 Geolocation

6.2.2 Ordnance Survey

OS Road (1 : 250 000), OS Landranger (1 : 50 000), OS Explorer (1 : 25 000)

6.2.3 Open Weather

6.2.4 Leaflet

Leaflet JavaScript library for maps

6.2.5 Mapbox

Mapbox script for GPX-GeoJSON conversion

6.2.6 Chart.js

Chart.js JavaScript library for ranger

6.3 Project Management

6.3.1 Life-Cycle & Timing

6.3.2 Supporting Tools

7 Evaluation

7.1 Requirements

7.2 Design, Construction & Implementation

7.2.1 Heuristic Evaluation

7.2.2 User Acceptance (UAT), Accessibility & Usability

7.2.3 ‘Prototyping’ & Constant Evaluation

7.3 Additional Testing

7.3.1 ‘Test-Driven’ Development

7.3.2 Unit Testing

7.4 Constrictions

Restricted to Glen Coe and Glen Etive

8 Development Conclusions

A Michael Mann Production
©1984 Orion Pictures Corporation
All Rights Reserved
Dolby Stereo™ In Selected Theatres

Appendices

Testing questionnaires, test cards (exp. result etc.), heuristic evaluation of system

Appendix 1: Existing Figures

Figure A1:

(Draper, Paudyal, 1999)

Figure A2:

(Draper, Paudyal, 1999)

Appendix 2: Foundational Material

Author's Note

The following content is manufactured by myself in aid of basic understanding of the background to contexts, data and methods present within this study. It is presented as teaching material, in a format I would output if in such position.

Figure B1: Linguistics of 'LaTeX'

LaTeX (or LaTeX, even latex (Donald E. Knuth's more recent installment of TeX)) is usually pronounced /lɑːtɛk/ ('lah') or /leɪtɛk/ ('lei'/'lay') in English (that is, not with the /ks/ pronunciation English speakers normally associate with X, but with a /k/). The characters T, E, X in the name come from capital Greek letters tau, epsilon, and chi, as the name of TeX derives from the Greek: $\tau\varepsilon\chi\nu\eta$ (skill, art, technique, precision); for this reason, Donald E. Knuth promotes a pronunciation of /tɛk/ (tekh) (that is, with a voiceless velar fricative as in Modern Greek, similar to the last sound of the German word "Bach", the Spanish "j" sound, or as "ch" in a Scottish 'loch').

Serif	**Sans Serif**	**Monospaced**
CMU Serif Roman	CMU Sans Serif	CMU Concrete
CMU Serif Bold	**CMU Sans Serif Bold**	
CMU Serif Italic		*CMU Concrete Italic*
CMU Serif Oblique	*CMU Sans Serif Oblique*	*CMU Concrete Oblique*
CMU Serif Small Caps		CMU Concrete Small Caps

In the presence of traditionalists, a suitable alternative to Donald E. Knuth's Computer Modern Unicode font family may be considered: Andale Mono.

Figure B2: Don. Knuth's Computer Modern Unicode (CMU) Font Family

Figure B3: Why My Pre-Title's Right and You're Wrong

I have received numerous comments which anyone would regard naïve and under-educated regarding my pre-title of this study: *AG436 Dissertation Coursework Assignment*. The argument originates in the 'Coursework Assignment' portion. People argue that a dissertation 'is not'/'does not have' an assignment. Not only is this poor characteristic recognition, it is semantically wrong. *AG436: Dissertation* is a class just like any other. However under this class, there are no lectures, no tutorials and therefore no exams as there is no [taught] content. Do not confuse this with the class 'having no content' though. AG436's content is apparent through literature of the student's choice. Therefore, it is possible for a 'coursework assignment' to be based on this. Hence, any further comments are null.

Bibliography

[1] Abraham, A., Ikenberry, D. L. (1994). *The Individual Investor and the Weekend Effect.* Journal of Financial and Quantitative Analysis. Volume 29. Issue 2. Pages 263-277.

[2] Investopedia. (2020). *What's the Most Expensive Stock of All Time?* Available At: `https://www.investopedia.com/ask/answers/081314/whats-most-expensive-stock-all-time.asp`. (Accessed: 08/10/2020).