Developing an Artificial Intelligence to combat inaccuracies within the weather forecasting sector

SCDT65 – Individual Project   
PIL17003689

Word Count:

# Contents

[Contents 2](#_Toc161396753)

[Abstract 3](#_Toc161396754)

[Ethical Approval 4](#_Toc161396755)

[Introduction 4](#_Toc161396756)

[Project Definition 4](#_Toc161396757)

[Project Aim 4](#_Toc161396758)

[Project Objectives 5](#_Toc161396759)

[Literary Review 6](#_Toc161396760)

[The Problem – Inaccuracies within the weather forecasting sector 6](#_Toc161396761)

[Forecasting Models 6](#_Toc161396762)

[Existing Solutions 7](#_Toc161396763)

[Medium Contradictions 8](#_Toc161396764)

[Existing Solutions 9](#_Toc161396765)

[Front-end Design and Development 11](#_Toc161396766)

[Web-Based Philosophies 11](#_Toc161396767)

[Justification of Tools 13](#_Toc161396768)

[Front End 13](#_Toc161396769)

[Back End 13](#_Toc161396770)

[Methodologies 16](#_Toc161396771)

[Design Methodologies 16](#_Toc161396772)

[References 18](#_Toc161396773)

# Abstract

Within the project, an artificial intelligence machine learning model with be developed using the Python programming language. Python’s support for libraries regarding data analysis and machine learning is uncontested regarding programming IDEs.

The dataset will be cleaned using the data analysis tools found within Jupyter Notebook’s libraries; NumPy, Matplotlib and Pandas. These libraries offer a multitude of data cleaning techniques to ensure the dataset is well suited for the AI.

Additionally, libraries with a heavy emphasis on machine learning algorithms, such as SciKit Learn, Pickle and Natural Language Toolkit to name a few will be incorporated to aid in the development of artificial intelligence.

Once the algorithm is finalised, it will need to be hosted on a webpage to be viewed. To achieve this, the micro web framework Flask will be used to run python scripts via a web solution, for example, on a PHP webpage.

# Ethical Approval

During development and testing, no ethical issues involving business or user data will arise. The dedicated dataset revolves solely around publicly available data within the past decade and will not involve any integration of personal information nor user interaction regarding personal data meaning ethical approval is unnecessary.

# Introduction

## Project Definition

In accordance with the Met Office (2020), the average weather forecast within the last decade has been criticised as potentially inaccurate with – on average – each report having an 80% chance to result in accurate predictions. Due to this inaccuracy, there is a 20% chance that a weather report will give false information to its users, resulting in vast user displeasure as said chance of failure can result in a multitude of issues regarding the user’s experience with and outside of the proposed solution.

To combat this, the solution will incorporate a wide dataset of weather forecasts within the past decade including the weather summary, humidity and temperature along with the use of a machine learning algorithm to identify patterns within the dataset and propose accurate predictions on weather forecasts.

Additionally, the web-based solution will provide great accessibility for users on multiple platforms including desktop and mobile devices.

## Project Aim

To develop and display an artificial intelligence that incorporates machine learning algorithms to predict current and future weather forecasts using a dataset through a web-based solution.

## Project Objectives

The project will achieve success through the following of these objectives:

* Commitment onto the literary review of the current topic.
* Data Analysis.
  + Commit to Data Understanding and Data Cleaning, two CRISP-DM phases.
* Design UX and UI.
  + Generate hypothetical user identity to pander design to.
  + Incorporate common web design laws into design i.e., Fitts’ Law.
  + Develop front-end UX and UI designs.
  + Perform Quality Assurance procedures to ensure the web solution is suitable.
    - Unit Testing.
    - Integration Testing.
    - Regression Testing.
    - Usability Testing.
    - Security Testing.
* Develop AI model.
  + Incorporate all CRISP-DM principles when deploying the model.
  + Perform supervised and unsupervised testing to create a model.
  + Supply model with datasets and perform predictions.
* Deploy AI model onto the web-based solution.
  + Incorporate the use of Flask and FastAPI to make the program recognise the AI as an API.
  + Integrate the API into the code using JavaScript.
  + Run continuous testing to ensure the API is fully functional on multiple devices.
* Showcase the software through GitHub repositories along with a server-side web host.
* Project Evaluation.

# Literary Review

## The Problem – Inaccuracies within the weather forecasting sector

### Forecasting Models

The weather forecasting industry sees an incline in efficiency and accuracy with each passing decade as predictions estimate an 80% average accuracy in 2021. (Klinger, K., 2021) This average has allowed predictions to span over a wider time frame. Since the weather itself is somewhat unpredictable, creating a report of estimates will decrease in accuracy the longer the estimation goes on for as it does not account for the current ongoing changes created by abnormalities like storms that would be easier to predict in a shorter time span. (Kealy, J., 2019)

The best theoretical solution would be to create new reports daily to ensure accuracy is at its peak which is what was carried out in the past. However, as the understanding of patterns grows and the incorporation of machine learning algorithms settle within the weather forecasting sector, predictions can occur between longer intervals as prediction models have improved. (Chantry, M., 2023)

One of these advancements within the sector is the Met Office’s Unified Model (2022)   
This model incorporates a hierarchy of configurations from a multitude of data reports from contributor such as the Joint Weather and Climate Research Program to summarise the most optimal weather forecasting results over different timeframes. I.e., A regional report from Euro4 will have a timeframe of 7 days and will cover a 4km distance, whereas a global report from MOGREPS-G will have a timeframe of slightly over 7 days but will cover a 10km distance. *(See Figure 1.)*

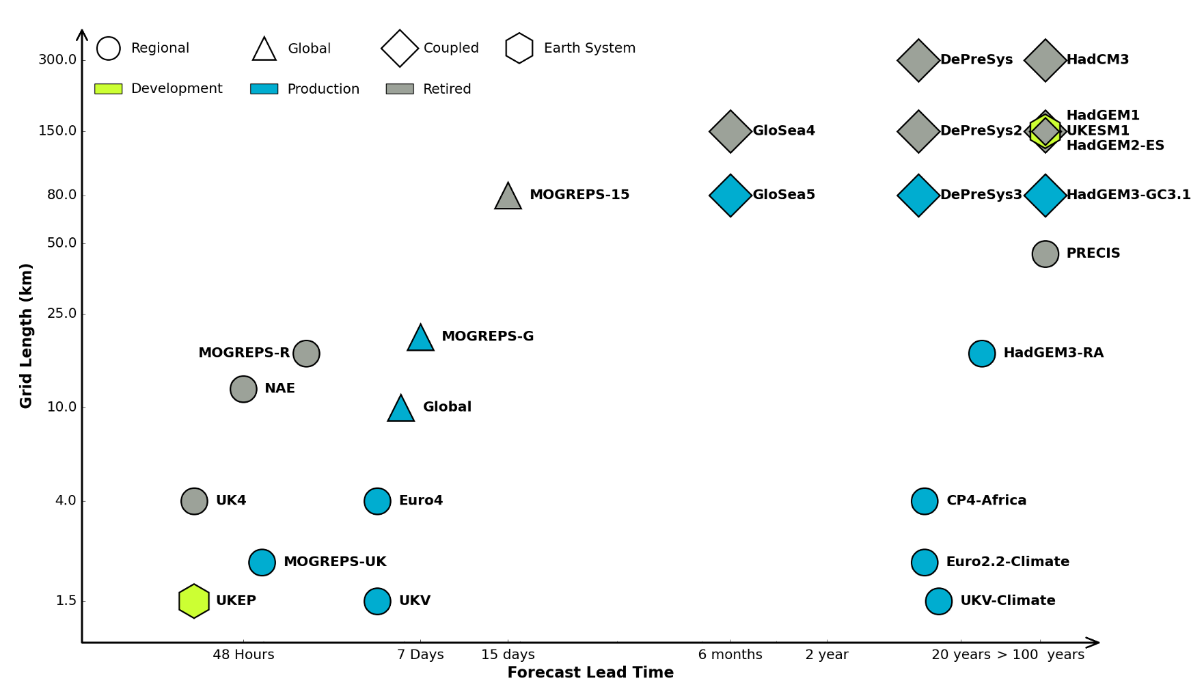


Figure 1: A diagram of the Met Office’s Unified Prediction Model. (2023)

This type of advancement within the sector has seen reports become undoubtably more accurate within the past decade, yet the results still receive criticism.

### Existing Solutions

Conventional weather forecasting methods resulted in moderately accurate results, but Google DeepMind provided recently that these models were severely inefficient. As the world incorporates artificial intelligence into more industries, such advancement is kept behind in the weather forecasting sector.



Source: [Medium.com](https://gordicaleksa.medium.com/how-i-got-a-job-at-deepmind-as-a-research-engineer-without-a-machine-learning-degree-1a45f2a781de)

DeepMind is an AI model made by Google to accurate predict forecasts using patterns that would not be discoverable by human interaction, going on to beat the world’s leading system in 90% of metrics used and taking only a fraction of time. (Cookson, C., 2023) The potential in AI development revolving around the weather forecasting industry has gone on to encourage firms to increase timeframes in which reports can be made, from the standard 5-7 days to 10 days whilst retaining the same accuracy.

DeepMind went further to predict weather anomalies, like Canada’s hurricane Lee 9 days before the event which was record breaking as usual models would have predicted the storm 3 days later. The machine-learning co-ordinator behind the case study at the European Centre for Medium-Range Weather Forecasts – or ECMWF – Matthew Chantry stated that “meteorological AI systems had progressed far sooner and more impressively than we expected even two years ago.” (Lam, R., 2023)

Additionally, associate professor of atmospheric science and hydrology at the University of Arizona, Kim Wood (2024) criticised existing AI solutions in a report to Scientific American, stating that “Because of their reliance on past data, most AI models might be poorly equipped to forecast rare and never-before-seen events.” Wood then proceeded to offer an example in “Hurricane Harvey, which dropped an unprecedented 60 inches of rain on parts of Texas in 2017.” This extremity could not be replicated by the behaviour of an AI as abnormalities akin to this will not be represented well in datasets. Furthermore, such extremities lack any form of long-term pattern recognition to identify these events.

To combat extremities in the past, researchers would identify patterns that would lead to rainfall and heavy storms by using short-range forecasts of 1 day, or sometimes less than a day, to improve accuracy tenfold (NOAA, 2021). These short-term forecasts will help researchers in spotting leads to an anomaly in the weather forecast and act upon it with haste. Such a practice is uncapable with unsupervised machine learning models and would instead force the idea of supervised classification or regression upon researchers when developing a model.

Concluding that artificial intelligence and machine learning, even within the past 5 years, had provided such an overwhelming advancement to weather forecasting despite few firms moving over to the new phenomenon and sticking to old mediums.

Though the model behind forecasting is not the only outdated problem within the industry, but also how data is presented.

### Medium Contradictions

A report from the BBC’s meteorologist, Nick Miller (2020) suggests that forecasting firms will not always report the same findings, stating; “If there is a 30% chance of a shower, one app might choose to show a shower symbol whereas another picks a partly cloudy symbol because there is a 70% chance it will stay dry.” Which brings up the lack of proper information displaying from firms within the sector.

This is due to medium of the forecast, where a user viewing the information on an app will tend to want faster data to intake that those on a television or desktop. A study conducted by the University of Alberta discovered that reading comprehension was impaired when content was presented on a mobile-size screen versus a larger computer screen. (Moran, K., 2016) This is due to the smaller text size and character amount in the smaller screen resolutions, so to combat this, apps will tend to focus more towards presenting bigger images or smaller data to the reader, making it easier to read.

When directly comparing the human brain’s ability to process data between images and text, according to The Media Education Centre (2010), we as humans process visuals 60,000 times faster than text, increasing the appeal of image-based sources due to the comprehensibility of the medium over a text-based solution. Furthermore, web publishers, on average, take in 12 times faster internet traffic growth due to their visual content than the average text-based web domain. (Alexis, C., 2018) This can be backed up by a study conducted by Brightcove Marketing that discovered that 95% of all respondents preferred image and video format over the typical text-based solution. (Kashyap, K., 2022)

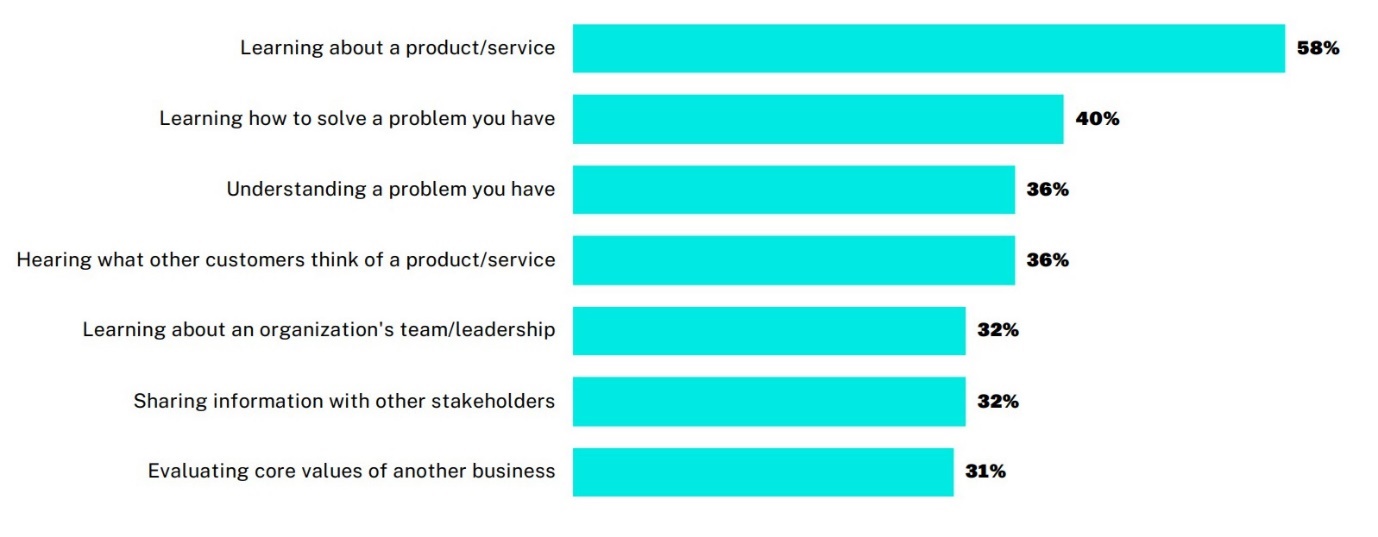


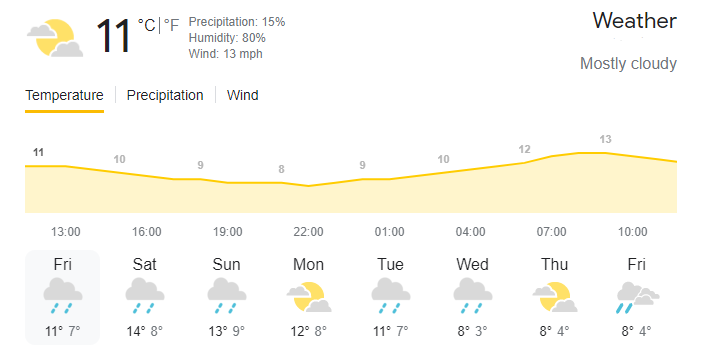
Figure 2: A visual display of how videos are most helpful when making a purchase decision.   
(Brightcove, 2022)

Figure 2 refers to how the respondents regarded the importance of video when displaying information with the most important factor being learning about the product/service at 58% with the second most important being learning how to remedy the problem that the product aims to solve at 40%.

With all of this in mind, the use of visual stimulus means that data is typically limited as to retain the user’s attention and, as per the previous example provided by Nick Miller, the implementation of imagery or video within the weather forecasting industry can obscure data to fit a graphic’s requirements, I.e., a firm that wants to display a 30% chance of shower as solely an image will tend to lean more towards the partly cloudy symbol as to use as little content as possible to display the information, whereas a more accurate reading by another firm may include text and other less visually appealing graphics to display more accurate information, overall losing out on user attention but displaying more accurate information.

### Existing Solutions

To combat this issue between offering accurate information and keeping viewer retention rates high to stimulate more traffic, a middleman must be met to fulfil both purposes.



Figures 3 and 4: Weather forecasts provided by Weather.com (3) and The Met Office (4).

These are two examples of successful weather report formats that are used today. Figure 3 follows a visual hierarchy regarding its layout, being the image in the top left and the chart in the middle. These two sets of data provide the user with a rough estimate of the forecast in the use of an image, the current temperature and a predictive chart regarding how the temperature will change throughout the day. Additionally, past and future weather forecasts are provided in a list at the bottom of the graphic with the same data as those in the visual hierarchy.

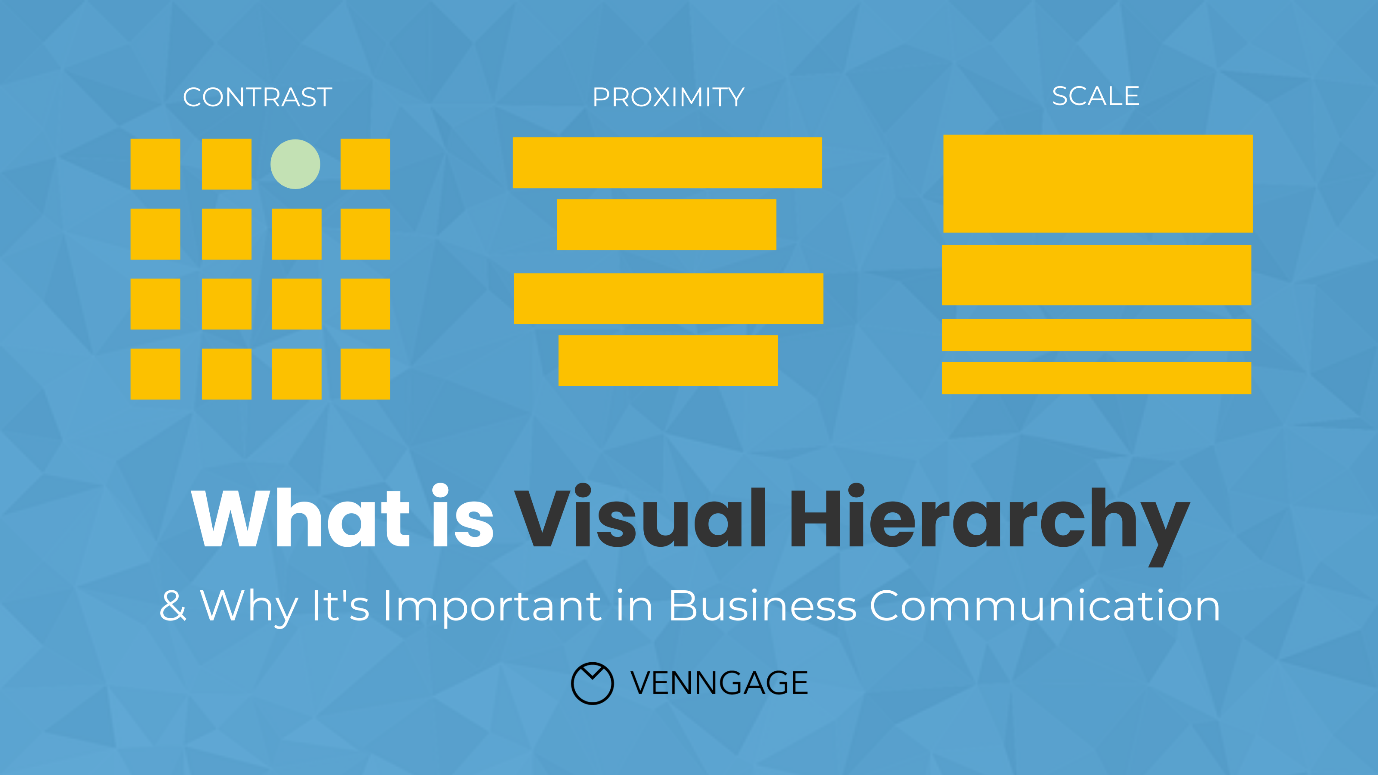


Figure 5: A representation of visual hierarchies. (Gaskin, J., 2022)

Figure 5 refers to the principles of a visual hierarchy, being; contrast – the identification of importance based on colour difference – proximity – the alignment of the content to represent how information is grouped together – and scale – the importance of the information reflected by how easy it is to see.

Judging by these measures, it is visible that Weather.com follows all three principles; the contrast between the image and chart being yellow in comparison to the black and white of the rest of the screen makes them stand out more, the proximity of the data fits as each day’s icon is allotted with information about what is forecasted, and the scale of the current day’s information is greater than every other day, with the icon and chart being the biggest which grabs the user’s attention.

However, sufficient data isn’t properly formatted in a quick and easy way for a user to see. In Figure 3, information regarding humidity, precipitation and wind speed is small and harder to see with the font colouring along with there is no information regarding aspects like the chance of rain or other details people may want to see leaving their interpretations down to an image which refers to the initial problem.

Figure 4 is another weather forecast provided by the Met Office. This figure provides a plethora of information that figure 3 lacks, including weather patterns during the day and UV and pollution levels. However, said figure does not abide by common web design laws, one of which being Fitt’s Law. This law indicates that the time taken to perform an action is dictated by the distance and size of the target object, making decisions slower if the target is small and far away and faster if the target is bigger and close by. (Guiard, Y., 2011)

Regarding figure 4, the data is relatively small and somewhat hard to see due to its size as well as the intractable buttons being relatively small compared to figure 3. The Met Office’s design focuses more on supplying the user with a wide assortment of information whereas Weather.com’s design focuses more on user intractability and ease of use, where one design succeeds, the other falls.

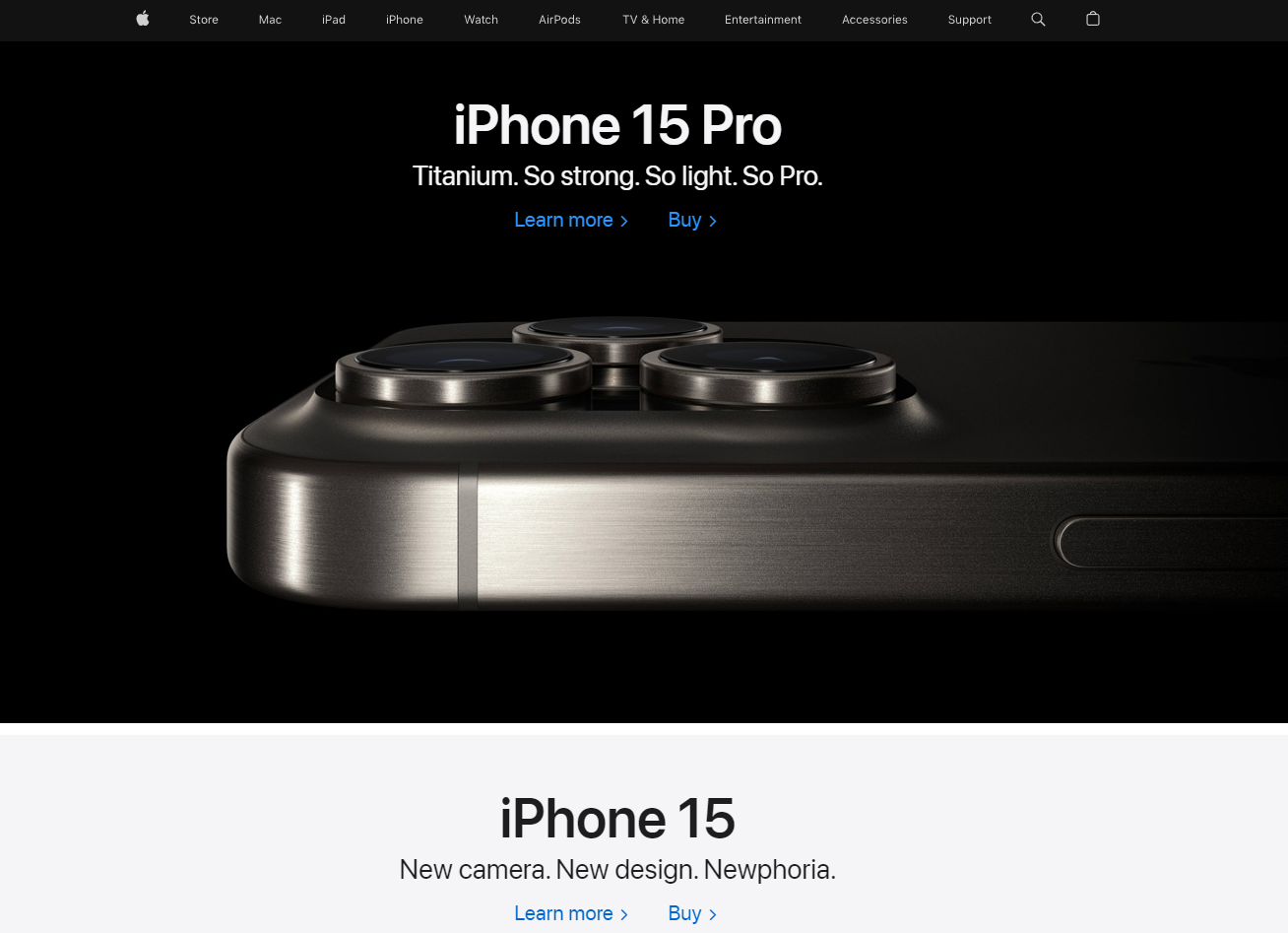
## Front-end Design and Development

### Web-Based Philosophies

As the project will take base on a web-based solution, the use of UX and UI design principles will improve user experience and boost the quality of the webpage. Such laws to follow consist of:

#### Fitts’ Law

A design law that states that the time required to move to a target is a function of the distance to the target and the size of the target (Laja, P., 2022). This relates to the relevance of an object being denoted to how easy it is to access. For example, a best-selling product on a shopping website will be displayed at the forefront of the webpage, making it the most interactable object the user can see, influencing their choice to interact with it.

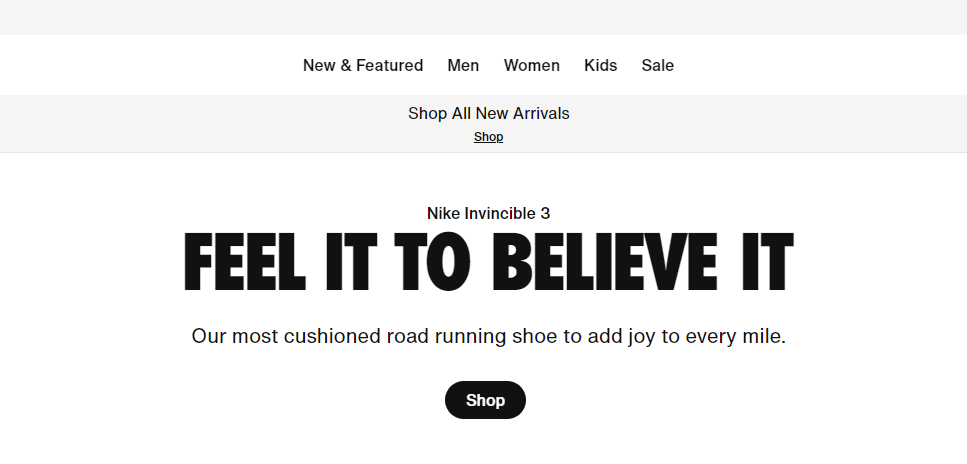


Source: [Apple.com](https://www.apple.com/uk/)

In the Apple website’s index lies an immediate presentation of their latest product, displayed in a majority of the screen, making it the most eye-catching object on the page. This example follows Fitts’ Law perfectly as the biggest and easiest to access target on the webpage is also the most important. Apple’s main incentive is to sell their latest product, so displaying it this way and following this design philosophy will achieve that goal more-so than if it was ignored.

#### Hick’s Law

Hick’s Law refers to the number of possible decisions equating to the time taken to decide. (Lavery, T., 2017) In this instance, if a web solution had multiple buttons within a singular navbar, the user’s choice to click on one would be amplified by the number of options available.



Source: [Nike.com](https://www.nike.com/gb/)

For example, the navigation bar of the NIKE website is minimalistic, providing enough options to fit the average viewer but also not overwhelming said viewer with variety within the first viewing of the website. Additionally, the focal-point of the page consists of the brand’s slogan long with a “Shop” button. As this button remains the only button on the webpage aside from the navbar, it fits the importance of Hick’s Law perfectly, being the only option, the viewer has, decreasing decision making time and overall increasing the efficiency of browsing the Nike online shop, which is the primary focus of the webpage.

* Visual Hierarchy
* Divine Proportions
* Rule of Thirds
* Gestalt Design Laws
* White Space
* Occam’s Razor

Tools

## Justification of Tools

### Front End

UX and UI Design

* + - PHP over HTML
    - Bootstrap

### Back End

#### Programming Language and IDE

For the project, Python found itself being the obvious choice of programming language when it came to developing a machine learning algorithm. Being critiqued as the most supportive of programming languages in the terms of machine learning development in accordance with IBM’s Tin Kam Ho (2019) and Telangana’s professor of computer science Srinath. (2017)

When validating which programming language to carry out the project with, it is crucial to include the context of the project within the conclusion. In this case, a direct comparison of IDEs can be made between Python and .NET during the justification of programming languages.

Python offers a vast number of libraries that can be used during the development and deployment of a machine learning model (Stančin, I., 2019), such libraries include; NumPy, which specialises in the optimisation of datasets, formatting them into arrays. The process of doing so speeds up the pre-processing of data within the data understanding and preparation phases of the CRISP data mining model and allows pre-processing functions to more effectively access the data due to the compact nature of an array, meaning less memory is allocated to storing the data, thus making such practices faster and less memory intensive. (NumPy, 2022) SciPy, which is an open-sourced library built off of NumPy that incorporates machine learning models into Python. These models consist of – but are not limited to - random forest, logistic regression and linear regression algorithms. (Virtanen, P., 2020) SciPy also offers sub packages that that specialises in the optimisation of models and error defining such as .optimize and .signal respectively. (SciPy, 2024) Finally, PyTorch is a third library within Python that offers tools regarding deep learning neural networking. These algorithms are used within high-level data management and wide-span projects. (Subramanian, 2018)

Many claims have conducted that Python is the best programming language in terms of compatibility when developing a machine learning algorithm (Muhammad, A., 2023). However, these claims can be contested, as Python retains vital downsides. Such disadvantages include a lack of universal back-end support

These libraries make Python a compatible programming language for a multitude of project types, spanning from big to small. However, this IDE still retains disadvantages. Such cons include a lack of official environment support within the context of the overarching brand. A feature that is greatly praised by its competitor, .NET (Choudhary, B., 2023).

.NET is a cross platform, open source application IDE produced by Microsoft (Regio, C., 2023), this means that it has full compatibility other Microsoft products. .NET excels with deploying windows-based applications and facilitates its machine learning library, ML.NET. The ability to co-operate with tools such as Unity and Tizen along with having inbuilt tools to make accessibility options, ML.NET makes itself a prominent option for developers.

These examples of .NET being used include SigParser, an email service that automatically generates and distributes non-human mail such as newsletters, sign up forms and password reset emails to those using the system. (Microsoft, 2013)

.NET also provides much more in-depth analysis and feedback when error handling, along with better performance along with performance diagnostics. This is achieved through .NET’s incorporation of just-in-time compilation within their IDE. JIT consists of the compiling – or translating of code to machine code – of the program being executed during the program’s runtime rather than before. (Aycock, J., 2003) This leans towards faster load times as the avoidance of compiling the entire application upon running means that compiling is reserved for when areas of code are called to be executed, contrarily increasing load time as areas within the program will need to be compiled when called whereas Python does not require anymore compilation after its initial batch, but this is negated by the fact that not all elements within an application will be used by a user, thus JIT ignores compiling areas of the program that would be unneeded to be compiled.

Ultimately, .NET’s better performance and support from other Microsoft products make it a valid choice when opting for a programming language in ML.NET. However, Python’s vast libraries and accessibility, with the likes of Jupyter allowing for easily executable model creations, avoiding the process of running an entire program and instead opting to execution certain factors of it, makes it the best choice for this project.

The justification behind this decision being that, although the features ML.NET provides are great, they are not as applicable in the context of the project as Python’s libraries. The support from other Microsoft products is useful for app development within Windows, though the nature of the project being a webpage makes this point redundant.

After the proposal of a programming language, the environment in which to develop on is the next factor to assess, two of the most popular IDEs in regards to machine learning development are PyCharm and Visual Studio Code, both of which support the Python programming language.

To begin with, a difference between PyCharm and Visual Studio Code – now being referred to as VSCode – is the nature of the IDE; PyCharm’s installations comes with prebuilt libraries, making it more of an “out the box” IDE (JetBrains, 2024). Due to the specialisation and focus on the Python programming language, PyCharm can afford coming preinstalled with the majority of Python libraries since it needs not to account for other languages (Kafesu, A., 2023), unlike VSCode. VSCode offers the support of many programming languages, including HTML, Java, C and Python, meaning developers must use multiple libraries to begin development. Because of this, VSCode can only come preinstalled with the bare minimum of language support to avoid large file sizes during installation. To combat this, VSCode incorporates the use of extensions. These extensions are third- and first-party libraries that can be installed directly from VSCode to add to a project. Extensions can range from library support for all the support languages, to problem and error handling using AI tools, and full customisation of the IDE, making it extremely user friendly (Chen, E., 2023).

VSCode also offers grander project management and deployment features in an inbuilt UI display for GitHub repository management, made even easier by the previously mentioned extensions system. This UI aids in making Git commits easier for those without terminal experience, or those who dislike the procedure of using a terminal (Speight, A., 2021). Meanwhile, PyCharm uses terminal based deployment through a command line interface after assigning a project to an account with their minimalistic UI. Though less beginner friendly, this approach is not necessarily a bad one, as the skills learned to use a CLI through the PyCharm IDE applies to other IDEs that use the same process such as other JetBrain developed IDEs like Atom whilst VSCode’s use of a UI tool is exclusive to other Microsoft products (GitHub, 2020).

This point also has area to change within the future as Microsoft’s acquisition of GitHub for ~$8.65 billion dollars (Warren, T., 2018) means more integration can be incorporated within other Microsoft products along with additional R&D being deployed for the GitHub windows application making terminal usage less and less used.

However, VSCode isn’t all praise. A study carried out by the North Carolina State University’s Elizabeth Lin (2024) concluded that there were a multitude of security risks involved with the incorporation of VSCode extensions within a project. This experiment consisted of testing the integrity of 43,436 extensions’ code with the likes of code injection attacks.

By conducting an analysis on how the extensions were built, malicious packages could be sent directly to the location of the file and thus gain access to information that is otherwise privatised. This is achieved due to the way in which extensions are made and processed. Extensions within VSCode are executed as child processes of the main VSCode application and built entirely from JavaScript. During the installation process of the extensions, information is provided to the user regarding the extension, such as the language ‘onLanguage’, the package name ‘package.json’, and its settings ‘settings.json’. From there, aspects of the extension such as handlers can be visualised and decrypted to reveal private information regarding the user and the author of the extension.

Additionally, another investigation regarding vulnerabilities within VSCode’s third party extensions was carried out by SonarSource’s Paul Gerste (2023). There, the team behind the investigation found similar exploits in the construction of handlers.

Within SonarSource’s testing, GitLens was one of the exemplar extensions that had this prominent issue. Safety measures within VSCode are in place to prevent any malicious code from being executed, an example being configuring handlers to default any untrustworthy areas of the program or extension as false. This is a form of input sanitization in an attempt to prevent any injection attempts from the user, such as the denial of code execution within input tags with the use of string conversion. However, practices like this can be overwritten by the developer when creating extensions within VSCode, which is how GitLens was exploited.

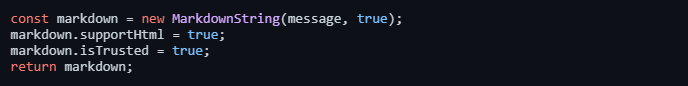
  
Figure 6: A handler within GitLens that trusts code written within a markdown section of the program. (Gerste, P., 2023)

Figure 6 displays an example of such thing happening, where the ‘markdown.isTrusted’ variable is set to True, meaning all input within a markdown text bot is executed without sanitization. From there, the user is free to perform malicious injections within the program such as forced installations.

Due to the unfiltered and unmonetized nature of VSCode’s extension library, creating and deploying extensions is easy for individuals to achieve and, once executed, a user is putting full trust into an extension over their IDE, which, due to the nature of the library, can result in potential security risks in regards to a user’s project, IDE or machine.

# Methodologies

A methodology is defined as the creation of multiple theoretical commitments about the intended outcomes of the investigation. (Howell, K.E., 2012) By this practice, the solution can be divided into thirds:

## Design Methodologies

Design methodologies or methods refer to the construction of new approaches in regards to problem solving within the nature of designing. (Cross. N., 1993) These methods span from data collection and research regarding the underlying issues the solution aims to solve, to deploying prototyping and applying extended design philosophies through the use of, for example, UX design.

For the project, two design methodologies were deemed most suitable; user-centred design and agile design. Both methods excelled in their application to the project in contrasting ways.

User-centred design – or UCD – consists of an emphasis on user feedback and response, satisfying the needs and preferences of the users of a product or service (Kumar, A., 2024). Within this design method, research is deployed in the medium of forms and questionnaires directed to the target audience in an attempt to extract as much, and as critical, user feedback as possible. This method aims to work upon features catered towards concise and specific feedback to appease the user as much as possible. Through the collection of constructive feedback, the strengths of the project can be honed along with the weaknesses being redefined, catering the project to the audience’s needs, having an emphasis on usability. (Kosinov, R., 2024)

From there, the method leans into the waterfall software development methodology as the provided user feedback allows the team to make full investments into long term changes with the new requirements (Dhandapani, S., 2016). As user feedback is specific, and the project well critiqued, the requirements provided during the UCD method are typically enough to offer a full waterfall cycle, making it the best fit design methodology for the software development methodology.

UCD is well applicable to large scale projects that have a vast number of clients (Chammas, A., 2015), this can be, for example, a hotel booking refurbishing project. As guests sign in, they can be given feedback mediums in the form of questionnaires to assess the current booking system and, after a specified time frame, feedback can be distributed to developers to take into consideration, creating a smooth flow of design and development.

In contrast, agile design – or AD – has better suitability on a lower range of stakeholders such as executive department heads and senior-level managers (Cooper, J., 2021). This design methodology focuses on applying agile methodology processes within the design phase of a project. These practices consist of rapid and constant prototyping based on requirements to create the most suitable solution to the issue. During this constant cycle of prototyping, the AD method applies constant cross-functional collaboration between the designers, developers, stakeholders and any other first and third parties associated with the project (Brush, K., 2022). Unlike UCD’s widespan feedback collection methods such as forms, AD can deploy more spontaneous feedback solutions in calls and meetings with stakeholders to assess the quality of the project (Wagener, B., 2023) and, with the agile nature of the method, these changes can be applied freely in contrast to UCD’s typically most compatible waterfall development methodology.

## Software Development Methodologies

An extension of comparisons between user-centred design and agile design can also be seen within the software development methodologies. First and foremost; software development methodologies are practices to ensure the development of information is very deliberate, structured and methodical, allowing the essence of development to be carried out rigidly and sequentially (Elliot, G., 2004).

Waterfall is an example of one of these software development methods. The method is approached linear, where each phase is executed and moved on from after completion. (Hoory, L., 2023) The methodology consists of five of these phases; requirements, design, implementation, testing and deployment.   
(See Figure 7)

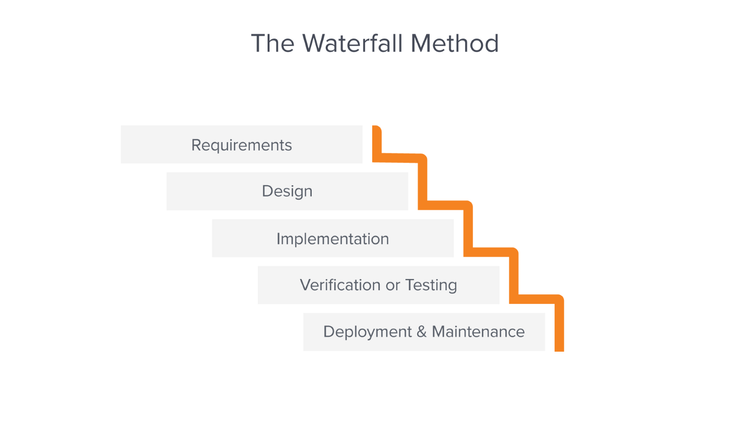


Figure 7: A visualisation of the waterfall SDM. (Lutkevich, B., 2019)

Once a phase is completed, it is deemed as completely finished, meaning there will be no regressing to previous phases during the cycle. (Hughey, D., 2009) This gives structure to the methodology as, by forcing teams to follow this regime, it ensures all areas of the project are completed to a high standard and all roadblocks are tackled immediately during their respective phases. (Ludidchart, 2017)

This professional state of structuration within the waterfall methodology has been recognised by firms within the real world, an example of such being the automotive manufacturer, Toyota. (Timeular, 2023)

In an interview with Toyota’s project general manager, Satoshi Ishii, he displayed a model of the company’s current development process using the waterfall methodology:

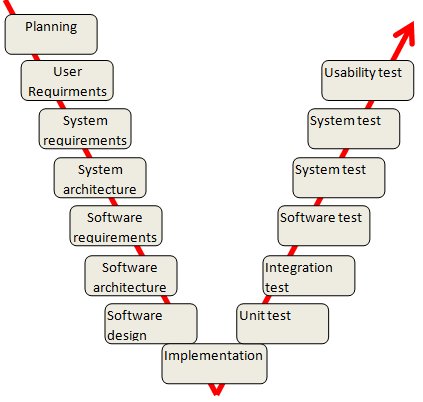


Figure 8: Toyota’s current development process. (Kniberg, H., 2010)

Figure 8 visualizes Toyota’s approach in design, with an emphasis on requirement identification and testing. As there is no repetition within the methodology, each phase must be thoroughly executed as to avoid any margin of error. The incorporation of multiple requirement and testing phases means more investigation can be put in place before and after the implementation of the product, reducing the overall margin of error.

However, though this methodology offers a well-structured work philosophy, it retains flaws. The process of a lack of repletion is flawed in of itself as testing is delayed, the implementation of test results cannot be deployed until the development phase is repeated within the methodology which can hinder a project in terms of time. (Paredes, R., 2023) Additionally, the methodology lacks any interaction with the client or user in terms of spontaneous feedback. Though there are user requirements, such requirements lack any proposals in regards to the current state of the project along with any changing opinions from the client, meaning a product may attain its assigned user requirements, it may not succeed in other aspects that need user feedback to prosper. (Singh, R., 2022)

The disadvantages surrounding the waterfall methodology are prevalent and have been a catalyst to make firms abandoning the method to pursue other methodologies. Toyota being one of these firms, with Satoshi Ishii stating that he was dissatisfied with how the design practices were affecting their means of production at Toyota. Such problems that were faced by Satoshi Ishii date back to the original concept of waterfall, with the very ideal of the methodology stating “The implementation described above is risky and invites failure.” When assessing the creation of the waterfall methodology. (Rovce, W., 1970)

With this in mind, Toyota would require an alternate methodology to accompany their transition, and the ideal choice for Satoshi was the agile methodology.

The agile methodology refers to a self-organizing approach to software development. (Collier, K.W., 2011) This means that, unlike waterfall, agile has an emphasis on repetitive development with constant researching and testing during “cycles.”

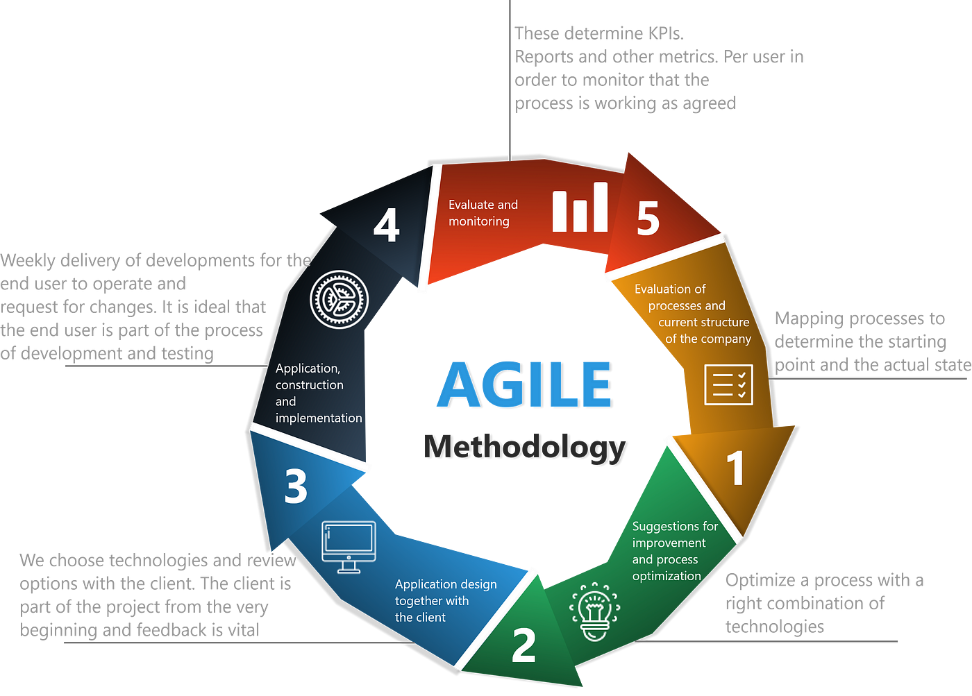


Figure 9: The Agile Methodology Lifecycle. (Abeythilake, U., 2022)

w

Testing Methodologies

# References

Abeythilake, U. (2022). *Agile Methodology*. [online] Medium. Available at: <https://medium.com/@abeythilakeudara3/agile-methodology-106270809c99>.

Alexis, C. (2018). *29 Incredible Stats that Prove the Power of Visual Marketing*. [online] movableink.com. Available at: <https://movableink.com/blog/29-incredible-stats-that-prove-the-power-of-visual-marketing>

Aycock, J. (2003). A brief history of just-in-time. *ACM Computing Surveys*, 35(2), pp.97–113. doi:https://doi.org/10.1145/857076.857077

Brightcove (2022) *The Power of Video in the B2B Buyer’s Journey.* [online] brighcove.com. Available at: <https://files.brightcove.com/WP_The_Power_Of_Video_in_B2B.pdf>

Brush, K. and Silverthorne, V. (2022) *What is Agile Software Development (Agile Methodologies)?* *TechTarget*. November 2022 [online]. Available from: <https://www.techtarget.com/searchsoftwarequality/definition/agile-software-development>

Chantry, M. (2023). *The rise of machine learning in weather forecasting*. [online] ECMWF. Available at: <https://www.ecmwf.int/en/about/media-centre/science-blog/2023/rise-machine-learning-weather-forecasting#:~:text=A%20revolution%20in%20ML%20models>

Chen, E., Huang, R., Chen, H.-S., Tseng, Y.-H. and Li, L.-Y. (2023). GPTutor: A ChatGPT-Powered Programming Tool for Code Explanation. *Communications in computer and information science*, pp.321–327. doi:https://doi.org/10.1007/978-3-031-36336-8\_50

Chammas, A., Quaresma, M. and Mont’Alvão, C. (2015) A Closer Look on the User Centred Design. *Procedia Manufacturing* [online]. 3, pp. 5397–5404. Available from: <https://www.sciencedirect.com/science/article/pii/S2351978915006575>

Choudhary, B. (2023). *Finoit Technologies*. [online] Finoit Technologies. Available at: <https://www.finoit.com/articles/python-vs-net-which-is-best-for-your-project/>

Collier, K.W. (2011). *Agile Analytics: A Value-Driven Approach to Business Intelligence and Data Warehousing*. [online] *Google Books*. Addison-Wesley Professional. Available at: <https://books.google.co.uk/books?id=0X7Dohne1tQC&newbks=0&hl=en&source=newbks_fb&redir_esc=y>

Cookson, C. (2023). *AI outperforms conventional weather forecasting methods for first time*. [online] Available at: <https://www.ft.com/content/ca5d655f-d684-4dec-8daa-1c58b0674be1>

Cooper, J. (2021). *Agile Project Management: A Guide to Stakeholder Updates, Alignment, and Transparency* [online]. Available at: <https://blog.zenhub.com/stakeholder-management-agile-software-development/>

Cross, N. (1993) *A HISTORY OF DESIGN METHODOLOGY* [online]. Available at: <https://monoskop.org/images/6/66/Cross_Nigel_1993_A_History_of_Design_Methodology.pdf>

Dhandapani, S. (2016). Integration of User Centered Design and Software Development Process. *2016 IEEE 7th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*. doi:https://doi.org/10.1109/iemcon.2016.7746075

Elliott, G. (2004). *Global Business Information Technology: An Integrated Systems Approach*. [online] *Google Books*. Pearson Addison Wesley. Available at: <https://www.google.co.uk/books/edition/_/qGfzMlfgzEcC?hl=en&gbpv=1&pg=PR13&dq=Geoffrey+Elliott+(2004).+Global+Business+Information+Technology>

Gaskin, J. (2022). *What is Visual Hierarchy & Why It’s Important in Business Communication*. [online] Available at: <https://venngage.com/blog/visual-hierarchy/>

Gerste, P. (2023). *Visual Studio Code Security: Markdown Vulnerabilities in Third-Party Extensions (2/3)*. [online] Available at: <https://www.sonarsource.com/blog/vscode-security-markdown-vulnerabilities-in-extensions/>

GitHub (2020). *Developing in a codespace*. [online] Available at: <https://docs.github.com/en/codespaces/developing-in-a-codespace/developing-in-a-codespace?tool=cli>

Guiard, Y. and Olafsdottir, H.B. (2011). On the Measurement of Movement Difficulty in the Standard Approach to Fitts’ Law. *PLoS ONE*, 6(10), p.e24389. doi:https://doi.org/10.1371/journal.pone.0024389

Hao, J. and Ho, T.K. (2019). Machine Learning Made Easy: A Review of Scikit-learn Package in Python Programming Language. *Journal of Educational and Behavioral Statistics*, 44(3), p.107699861983224. doi:https://doi.org/10.3102/1076998619832248

‌

Howell, K.E. (2012). *An Introduction to the Philosophy of Methodology*. [online] *Google Books*. SAGE. Available at: <https://books.google.co.uk/books?id=uYkRAgAAQBAJ&printsec=frontcover&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false>

Hoory, L. (2023) *What Is Waterfall Methodology? And Can It Help Your Project Management Strategy?* *Forbes Advisor UK*. 7 July 2023 [online]. Available from: <https://www.forbes.com/uk/advisor/business/what-is-waterfall-methodology/>

Hughey, D. (2009). *The Traditional Waterfall Approach*. [online] Umsl.edu. Available at: <https://www.umsl.edu/~hugheyd/is6840/waterfall.html>

JetBrains. (2024). *Built-in Developer Tools - Features | PyCharm*. [online] Available at: <https://www.jetbrains.com/pycharm/features/tools.html>

Kafesu, A. (2023). *VS Code vs. Pycharm: The Best IDE for Python*. [online] Geekflare. Available at: <https://geekflare.com/vs-code-vs-pycharm/>

Kashyap, K. (2022). *95% B2B Buyers Say Videos Play Crucial Role in Their Purchase Decisions*. [online] Available at: <https://www.spiceworks.com/marketing/content-marketing/news/b2b-buyers-say-videos-play-crucial-role/>

Kealy, J. (2019). *Long-range weather forecasting: the chaos of the 10-day forecast*. [online] Available at: <https://www.yourweather.co.uk/news/science/long-range-weather-forecasting-the-chaos-of-ensembles.html>

Klinger, K. (2021). *How reliable is your weather forecast?* [online] Available at: <https://www.agriculture.com/weather/news/how-reliable-is-your-weather-forecast>

Kumar, A., Kosinov, R., Pavlova, O. (2024). *How do you provide clear and meaningful feedback in user-centred design?* [online] Available at: <https://www.linkedin.com/advice/1/how-do-you-provide-clear-meaningful-feedback#:~:text=Feedback%20is%20an%20essential%20part,and%20communicate%20your%20value%20proposition>

Laja, P. (2022) *8 Web Design Principles and Laws that work in 2023.* [online] CXL. Available at: <https://cxl.com/blog/universal-web-design-principles/#h-4-fitt-s-law>

Lam, R., Sanchez-Gonzalez, A., Willson, M., Wirnsberger, P., Fortunato, M., Ferran Alet, Suman Ravuri, Timo Ewalds, Zach Eaton-Rosen, Hu, W., Merose, A., Hoyer, S., Holland, G.A., Oriol Vinyals, Stott, J., Pritzel, A., Mohamed, S. and Battaglia, P.W. (2023). Learning skillful medium-range global weather forecasting. *Science*. doi:https://doi.org/10.1126/science.adi2336

Lavery, T., (2017). *What is Hick’s law? - Definition from WhatIs.com*. [online] Available at: <https://www.techtarget.com/whatis/definition/HiWcks-law#:~:text=Hick>

Leffer, L. (2024). *AI Weather Forecasting Can’t Replace Humans--Yet*. [online] Scientific American. Available at: <https://www.scientificamerican.com/article/ai-weather-forecasting-cant-replace-humans-yet/>

Lin, E., Koishybayev, I., Dunlap, T., Enck, W. and Kapravelos, A. (2024). UntrustIDE: Exploiting Weaknesses in VS Code Extensions. *Proceedings 2024 Network and Distributed System Security Symposium*. [online] doi:https://doi.org/10.14722/ndss.2024.24073

Lucidchart. (2017). *The Pros and Cons of Waterfall Methodology*. [online] Available at: <https://www.lucidchart.com/blog/pros-and-cons-of-waterfall-methodology#:~:text=Advantages%20of%20the%20Waterfall%20model>

Lutkevich, B. (2019) *Definition: Waterfall model*. [online]. Available from: <https://www.techtarget.com/searchsoftwarequality/definition/waterfall-model>

Met Office. (2020). *Global accuracy at a local level*. [online] Available at: <https://www.metoffice.gov.uk/about-us/what/accuracy-and-trust/how-accurate-are-our-public-forecasts>

Met Office. (2022). *Unified Model*. [online] Available at: <https://www.metoffice.gov.uk/research/approach/modelling-systems/unified-model>

Moran, K. (2016). *Reading Content on Mobile Devices*. [online] Nielsen Norman Group. Available at: <https://www.nngroup.com/articles/mobile-content/>

Microsoft. (2013). *SigParser uses ML.NET to detect ‘non-human’ emails | .NET*. [online] Available at: <https://dotnet.microsoft.com/en-us/platform/customers/sig-parser>

Miller, N. (2020). *BBC News*. [online] 18 Feb. Available at: <https://www.bbc.co.uk/news/explainers-51533852>

Muhammad, A. (2023). *Is It Worth Learning C# and .NET Technology in 2023-24? A Comparative Analysis with Python*. [online] Available at: <https://www.linkedin.com/pulse/worth-learning-c-net-technology-2023-24-comparative-python-abubakar#:~:text=NET%20or%20Python%20in%202023>

NOAA National Severe Storms Laboratory. (2021). *Thunderstorm Forecasting*. [online] Available at: <https://www.nssl.noaa.gov/education/svrwx101/thunderstorms/forecasting/#:~:text=NSSL%20develops%20ensembles%20for%20very>

NumPy (2022). *NumPy: the absolute basics for beginners — NumPy v1.20 Manual*. [online] numpy.org. Available at: <https://numpy.org/doc/stable/user/absolute_beginners.html>

Paredes, R. (2023). *Waterfall Methodology: The Pros and Cons*. [online] SafetyCulture. Available at: <https://safetyculture.com/topics/waterfall-methodology/>

Regio, C. (2023). *What is .NET? (Part 1 of 3)*. [online] learn.microsoft.com. Available at: <https://learn.microsoft.com/en-us/shows/dotnet-for-beginners/what-is-dotnet-dotnet-for-beginners>

Rovce, W. (1970). *Managing the Development of Large Software Systems*. [online] Available at: <https://leadinganswers.typepad.com/leading_answers/files/original_waterfall_paper_winston_royce.pdf>

SciPy (2024). *Search — SciPy v1.12.0 Manual*. [online] Available at: <https://docs.scipy.org/doc/scipy/search.html?q=Introduction>

Singh, R. (2022). *Waterfall Methodology*. [online] Institute of Project Management. Available at: <https://instituteprojectmanagement.com/blog/waterfall-methodology/>

Srinath, K.R., (2017) *Python – The Fastest Growing Programming Language.* [online] Available at: <https://www.irjet.net/archives/V4/i12/IRJET-V4I1266.pdf>

Speight, A. (2021). *Visual Studio Code for Python Programmers*. [online] *Google Books*. John Wiley & Sons. Available at: <https://books.google.co.uk/books?hl=en&lr=&id=DdgxEAAAQBAJ&oi=fnd&pg=PP23&dq=vscode+extensions+python&ots=q4PQfsSLY_&sig=ip7qbFeDvi3m7ePDLgt3-CL6UrU&redir_esc=y#v=onepage&q=vscode%20extensions%20python&f=false>

Stančin, I. and Jović, A. (2019). *An overview and comparison of free Python libraries for data mining and big data analysis*. [online] IEEE Xplore. doi:https:// doi.org/10.23919/MIPRO.2019.8757088

Subramanian, V. (2018). *Deep Learning with PyTorch: A practical approach to building neural network models using PyTorch*. [online] *Google Books*. Packt Publishing Ltd. Available at: <https://books.google.co.uk/books?hl=en&lr=&id=DOlODwAAQBAJ&oi=fnd&pg=PP1&dq=neural+networking+using+pytorch&ots=kpY1l3hBPb&sig=826MgARR1HcgP8JRXfvLhFi3hL8&redir_esc=y#v=onepage&q=neural%20networking%20using%20pytorch&f=false>

The Media Education Centre (2010) *Williams College*.[online] Available at: <https://oit.williams.edu/files/2010/02/using-images-effectively.pdf>

Timeular (2023). *Waterfall Project Management Methodology*. [online] Timeular. Available at: <https://timeular.com/blog/waterfall-project-management-methodology/#:~:text=The%20most%20recognized%20company%20that%20uses%20the%20Waterfall%20methodology%20is%20Toyota>

Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S.J., Brett, M., Wilson, J., Millman, K.J., Mayorov, N., Nelson, A.R.J., Jones, E., Kern, R., Larson, E. and Carey, C.J. (2020). SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods*, 17(3), pp.261–272. doi:https://doi.org/10.1038/s41592-019-0686-2

Warren, T. (2018). *Microsoft completes GitHub acquisition*. [online] The Verge. Available at: <https://www.theverge.com/2018/10/26/17954714/microsoft-github-deal-acquisition-complete>

Wagener, B. (2023). *The Importance of User Feedback for Product Development in Agile Environments* [online]. Available at: <https://www.linkedin.com/pulse/importance-user-feedback-product-development-agile-environments#:~:text=Unlike%20traditional%20methodologies%2C%20agile%20product,are%20most%20valuable%20to%20users>