

ST10084595

Programming POE

Part 1

PROG7311

Introduction

The company that has been approached is known as Olympian Software. This IT company specialises in the development of websites.

The company that approached them is known as Farm Central. They have requested that the website can track all the incoming stock as well that is outgoing stock.

The website will contain the ability to calculate the amount of stock that Farm Central has, it will be able to increment or decrement the amount of stock that is coming in or being used to make their products. The website will also display which farm the stock is coming from, as well as which farmer owns the stock as there might be multiple farmers who own the farm. The site will be easily useable. It will have a side menu panel that will display all the stock and categorise it accordingly. Inside each of the categories, it will arrange the stock in alphabetical order. Each item will have a picture of stock. Upon opening an item, it will display all the item information being its description, what it is used for, as well as price. The web site will be user friendly to ensure that the customers and the client for this website, will not be overwhelmed with information and a confusing website.

The purpose of this website is to ensure that a stock supplier for a vast number of farms can easily track, record, and ship their stock to the various farmers that have ordered from them. The website will track all outgoing orders to their clients.

Upon arrival to the client, they will receive an email and or tracking information about their packages. The managers of this website will be able to list and take down items at will. Only they will have that authorisation to do so. This is to ensure that the customers will not have access to any of the supplier's stock. The software will make use of a network pattern known as MVC's (Microsoft, 2023). This type of architectural pattern in software design is mainly used to implement the user interfaces. Data, and to control the overall logic of the website. This pattern

emphasizes separation between the software's business logic, as well and the display of the website (Microsoft, 2023).

The website will also make use of a Login and a Registration page. The login and registration will all make use of hashing the passwords for each account to ensure that each person's data is secure to avoid being hacked (Insights, 2023). The website will also send an email to the user when they are registering an account to ensure that it is the set user creating the account (Insights, 2023).

The website will dawn a colour scheme that is not to flashy and hard on the eyes. It will be a user-friendly colour scheme as everyone reacts or may react differently to bright colours.

The sites design will have frontend code to give it that extra step to looking more presentable. It will have different stylings to have certain parts of the site stand out from the rest.

On each button, there will be a hover instance that highlights the button into a different colour as well as give it a different texture that is different to the rest.

Non-Functional Requirements

Performance

Web performance – This is the speed at which the sites content loads and renders inside of a web browser, as well as how responsive it is to user interactions. With regards to sites that perform poorly, they are slow to respond to input and slow to display content (Foundation, The "why" of web performance, 2023).

It is important to have good performance for a website as it ensures that a user has good experience while interacting with it. In a matter of sorts, the performance of a website may be considered as a subset of web accessibility. With performance, you take into consideration what type of device the site is being visited on (laptop, phone) as well the devices connection speed.

- To overcome performance issues and to improve the performance of the website, optimizing image sizes is an effective way to do so. Many sites are heavily reliant on graphics such as pictures and or videos, so if the site contains graphics that are of too high resolution, then it will cause the site to slow down (Foundation, The "why" of web performance, 2023).

EG: If a website is being loaded using a high-resolution display screen, then it will have no trouble rendering the high-definition pictures and videos, but, if the screen being used is not high quality, then it will increase the bandwidth used and slow down the site.

- Another way to optimise performance is to minimize time to first byte. This is the time it takes for a browser to receive the first byte of data from a server. To implement this fix, upgrading the server's hardware to have more memory or CPU is recommended (Foundation, The "why" of web performance, 2023).

With regards to site development, the images and or videos that are of recommended use should be that of low or medium quality in order to speed up load time, as well as reduce the bandwidth of the site (Foundation, The "why" of web performance, 2023).

Reliability

The reliability of a website is determined by whether the image and video quality are at adequate resolution to ensure that the site is not “Hard on the eyes.” Reliable websites tend to contain images that are of a resolution that captures a viewer’s attention.

- Regardless of what the site is loaded on, it looks good.

The design of the website is also a primary contributor to the judgement of its reliability. A site that’s well-designed shows users that the company is one of seriousness and have put in an investment in themselves to have a presentable website. If a site is poorly designed, has sloppy writing upon it, then it is unreliable.

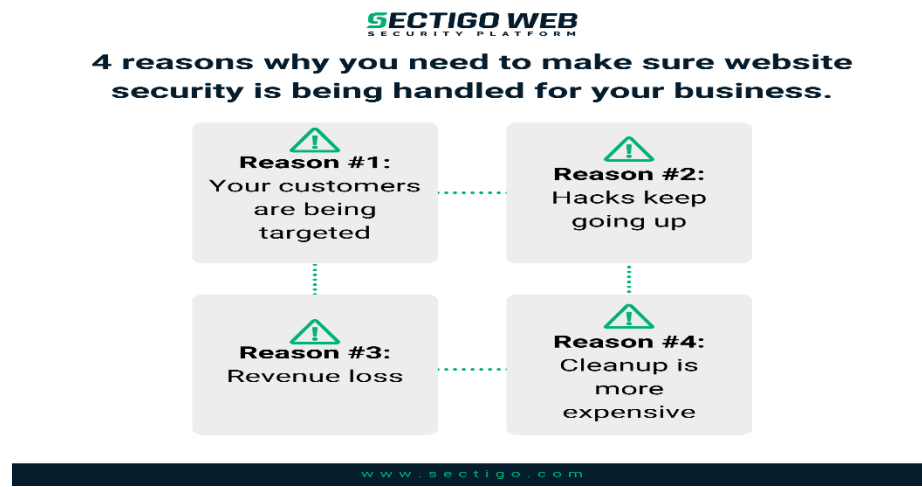
To ensure reliability of this site, there will be a use of images that are of a quality that captures a user’s attention regardless of the device they are accessing it from. With regards to the information on the site, they will be taken multiple sources in order to solidify its reliability.

The design of the site will contain styling code that will control all the parts of the site when the cursor is moving around it. The buttons will have hover functions so that they change colour as well as textures to attract attention, and to show the user which section they are about to enter.

The styling parts to the website adds to the reliability of it as it does not affect the load time of the site but it adds more professionalism (Foundation, CSS: Cascading Style Sheets, 2023).

The styling of the site is known as the CSS. The purpose of CSS is to style and layout web pages. CSS stands for Cascading Style Sheets (Foundation, CSS: Cascading Style Sheets, 2023). CSS describes how the elements of a website will be displayed on a live screen (Foundation, CSS: Cascading Style Sheets, 2023). The purpose of using CSS is that it saves a lot of work, as well as it controls the layout of multiple web pages all simultaneously (Foundation, CSS: Cascading Style Sheets, 2023).

Security



(Sectigo, 2023)

The importance of security for a website is to ensure that hackers as well as cyber-thieves do not gain access to sensitive information.

In order to check the security of a website, the implementation of HTTPS must be used.

- Improving the security of a site would include the use of strong passwords. (UpGuard, 2023)(Sectigo, 2023).
- Two factor authorisation makes it easier when a site recognises an IP address being used to login to a website is different. The user will immediately be sent a text to confirm that they authorise this login attempt (UpGuard, 2023)(Sectigo, 2023).
- The use of secure networks is also recommended when logging into financial and other crucial websites. To identify this, look at the link tag, it should start with HTTPS to ensure that it is a secured network (UpGuard, 2023)(Sectigo, 2023).

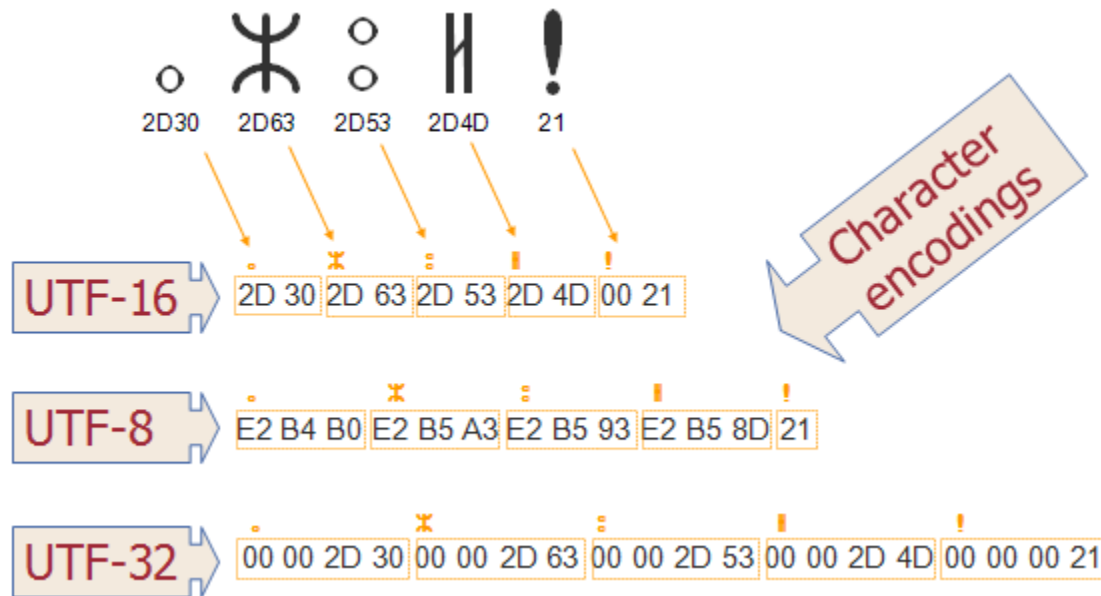
With regards to the security of the website, before registering an account the user will be sent an email to confirm that it is them creating the account. The user's password will also be encrypted when it is being pushed into a database for storage. The information that the user enters the website, will only be accessible by that same user. No other user will be able to access other users' information unless they permit so (Sectigo, 2023).

Upon login for the website, if the user has forgotten his/her password then they will be able to reset it using their email address. There will be a link on the login page.

When the user creates a password, the password will then be hashed into the UTF-8 encryption method.

There are 3 main methods for encrypting a password or a set of information.

- Symmetric- This type of encryption method uses a single key to do both encrypt and decrypt a set of data.
- Asymmetric- This encryption method is also known as public key encryption. This encryption method is much more secure than Symmetric as there is no reason to share the decryption key. It also simplifies the key management. With this encryption method, the user will need to generate a new key for each contact to ensure security.
- Hashing- this involves the replacement of contents or the replacement of a summary of a file with a value that has a value of a fixed-length which is based on a mathematical formula. The key feature of hashing is that once data is encrypted, then it is not intended to be decrypted. This should be an irreversible process.



(javarevisited, 2023)

The method of choice for this encryption method will be Hashing. The hashing method that will be used will be UTF-8 (StackExchange, 2023).

The UTF-8 encryption method is an encoded system for Unicode. It has the capability to translate and Unicode character into a unique binary string to match it. It also can translate the binary string back into the Unicode character. The term UTF means Unicode Transformation Format (StackExchange, 2023).

The purpose of UTF-8 is that it represents the characters as ASCII text while still granting access for international characters (Chinese Characters.) (StackExchange, 2023).

ASCII is a code set which contains 128 code points. This specific code set contains punctuation marks, digits, control characters, and it also has the uppercase and lowercase English alphabet (StackExchange, 2023).

Usability

Usability is how well a user can navigate through a website without being overwhelmed by too much information or disappointed by too little. The usability of the site should be “easy on the eyes” but not too overwhelming that they get tired of visiting it. The website should be easily navigational, and easy to use.

- The site should have dedicated sections to group different types of items together.
- As well as have different sections to keep the site user friendly and easily accessible.
- The user should not have to spend a long time trying to find certain information or try to navigate to a specific section in a long period of time.

This site will implement a menu bar that will navigate to different web pages for easy access to the different criteria that will be met for the website.

Every aspect of the site will be associated with a picture to show the user what that specific part of the site is doing. For e.g: a shop page will have a shop icon. Each element will have its own unique icon. When a user is to hover over the icon, the site will display the name of that section as well.

List of items in stock

All the items that have been displayed onto the site will have a search bar above it to navigate through different item names as well as their different categories. On the items in the site, there will be a running total (stock count) of the amount of stock there is per item. If an item is out of stock, the website will either remove the item from the site entirely or it will give an option to put in a pre-order for the item so that when stock is available, the people who have pre orders will be the first to get it. On the items listed page, when the user opens an item, it will then show a full run through of the item. It will show the items name, description, stock count, as well as the items intended use purposes. The site will also have a picture of the item.

All the items that will be shown onto the site will be arranged in either alphabetical order, price highest to lowest, or stock quantity from highest to lowest, or the user can look for out of stock items so they can arrange for a pre-order.

The items that have been listed will show the amount of stock they have of that item, whether there is a sale on the item, and whether the item is running out of stock or not.

When the item stock count gets to a certain quantity, eg:10, the website will then change the colour of the quantity text to a red colour to indicate that the stock is almost gone so that it captures the user's attention.

The listing of items will show the specific type of item that it is, whether it is animal feed, or basic farming tools. The site will categorise each item into its respective sections to simplify the process and increase user friendly status as it will make the website easier to navigate through.

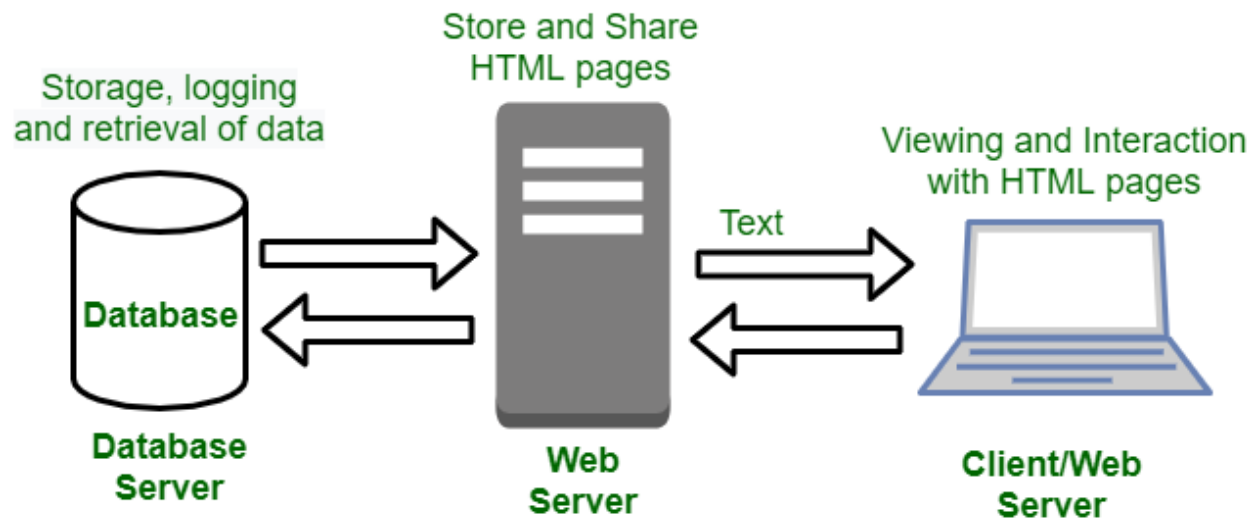
Overall site explanation

This website will make use of MVC's that will be designed and coded in visual studio 2022. With regards to the usability and the overall design of the website, it will make use of a menu panel that will allow for easier grouping of items as well as easy navigational purposes for the user. The website will make use of a user login system as well to ensure that all the data permitted to the site is saved under each separate person so that no one can view each other's information. User specific data. Upon creation of the account to access the website, the user will receive an email asking them if they wish to accept this registration. The website will have pictures that are of a resolution to ensure that the reliability to the site is valid as well as takes into consideration the bandwidth of the site. The resolution of the images and videos should not hinder the performance of the site.

Upon the creation of an account, each user's password will be saved as its own unique string to prevent and security breaks(hackers). All the user information will be pushed into a database. The website will make use of blob storages, so therefore payments will occur as they will be stored to the azure cloud. The reason for using blob storage, is that the website will contain pictures and or videos, so therefore they will need to be stored into the database as well. The blob storage will be a pay as you go type of website as it will make use of the amount of people are using the site, as well as the amount of item that are displayed. The higher the amount of pictures and items on the site, the higher the fee (Cloudflare, 2023).

The website will make use of a forgot password section. The user's profile will contain a password change area.

When a user makes a purchase from the Farm Central site, they will receive tracking info about their package's whereabouts.



(GeeksForGeeks, 2022)

This is an example of a website storing its information into a database.

On the right-hand side is a client server. That is where the website will be viewed and or edited. In the middle of the diagram is a web server. This web server stores all the information that is being relayed to and from the database, as well and information to and from the client server.

On the left-hand side is the database. That is where all the information is going to be stored (encrypted passwords, emails, usernames, the user's information).

Are Design And Architectural Patterns Relevant?

Design and architectural patterns are important for a website as they improve the efficiency, the productivity, and the speed of the site. They are also important in the development of costs, as well improve the planning of the website (Hat, 2023).

Architectural Patterns

This focuses more on the abstract view of an idea. Generally, an architectural pattern is a reusable solution to an occurring common problem inside of software architecture within a given context (Simform, 2023).

Architectural patterns address numerous different issues in that of software engineering, eg: the performance limitations of computer hardware, as well as the availability and minimization of a business' risk.

Examples of architectural design:

- MVC pattern (Simform, 2023).
- MVVM (Simform, 2023).
- N-tier (Simform, 2023).
- Message bus (Simform, 2023).
- Microservices (Simform, 2023).
- Domain-driven design components (Simform, 2023).
- Presentation-abstraction-control (Simform, 2023).

The importance of architectural design is that these software patterns drive efficiency and productivity towards the project. A developer may join any existing project at any time with limited onboarding as they are fully aware of the patterns being used in the projects (Simform, 2023).

Design Patterns

This focuses on the implementation of an idea. In that of software engineering, a design pattern is a general, reusable solution in order to solve a common problem when designing an application system. A design pattern is more similarly like that of a template in order to approach a problem at hand, unlike that of a library or a framework (Simform, 2023).

Examples of design patterns:

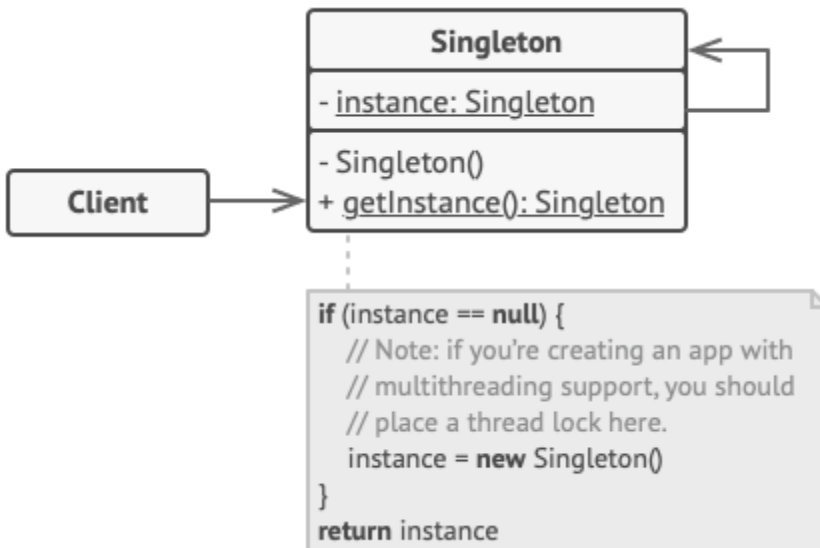
- Singleton Design Patterns (Simform, 2023)
- Façade Design Patterns (Simform, 2023)
- Factory Design Patterns (Simform, 2023)
- Strategy Design Patterns (Simform, 2023)
- Observer Design Patterns (Simform, 2023)
- Builder Design Patterns (Simform, 2023)
- Adapter Design Patterns. (Simform, 2023)

Design patterns are useful as they accelerate the development process. Design patterns provide proven development paradigms; this is a time saver as it does not reinvent the patterns every time a problem is at hand.

The creation of usage of patterns encourages software reuse. A pattern is used multiple times, but is only designed once. The reuse of patterns causes the production cost to drop, as well as it saves time by reducing the time it takes to redesign. The reuse of patterns creates higher reliability in the software, as well as causes higher continuity to the code design.

Singleton Design Pattern

The design pattern that will be used for this website will be Singleton design patterns.



(refactoring.guru, 2023)

The use of this design pattern is for logging, driver's objects, caching, and thread pool. Singleton is also used in multiple other design patterns such as Abstract Factory Builder, Prototype, and Façade.

Singleton Design Pattern involves a single class that is responsible in the creation of an object while it ensures the creation of a single object. The purpose for this class is to provide a different way in order to access its only object which can be accessed without the need to instantiate the classes object (tutorialsPoint, 2023).

There are 2 types of this design pattern (betterProgramming, 2023):

Eager initialization.

The singleton object is pre-initialized upon the creation of the class (SpringHow, 2023).

This is recommended for smaller objects. Inflation of the startup tie would be initialized on large objects (SpringHow, 2023).

Lazy initialization.

Singleton object only when it is required by another class (SpringHow, 2023) .

The same object will be returned in order to be requested in the future (SpringHow, 2023).

The difference between them is that they initialize the singleton instance. Thread-safety is also considered between the two types.

The singleton design pattern should be used when a class in the program has only a single instance available to all clients. This pattern disables every other instance of creating the objects of a class, except the special creation method.

This design pattern in the use of a website will prevent multiple instances from being active simultaneously as it can cause there to be bugs in the site. This mainly gets implemented inside of the constructor. The reason for using singleton patterns is for the regulation of the global state of the application.

Advantages of Singleton.

Singleton is a worldwide object in disguise however, it provides a global access point. The singleton design pattern is accessible from anywhere in the program but this can not be modified from anywhere, it has to be modified inside of the singleton. This design pattern is a mean to protect global (digitalOcean, 2023).

A major advantage of singleton is that it is a huge memory saver as the object is not created at every single request. Single instances are only reused over and over (modernescpp, 2022).

- Global Access Point (modernescpp, 2022).
- Unique Entity Model (modernescpp, 2022).

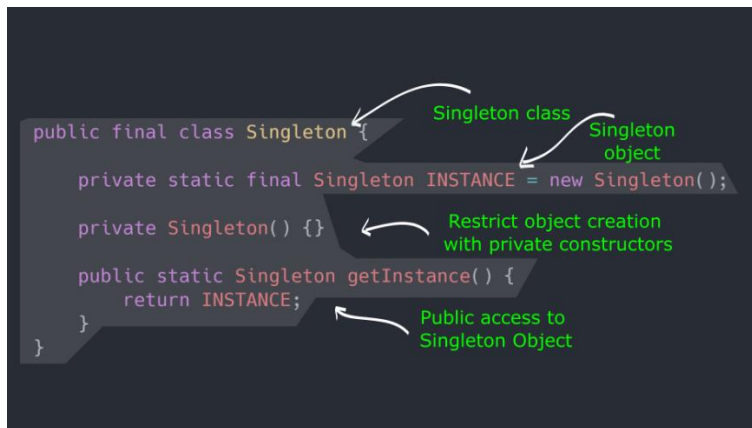
Compared to global variables, the instance fields of singleton do not occupy space in the global namespace. Singletons also can be initialized lazily.

Over static classes, singletons hold an instantiated object. Advantages that singletons have over static classes are:

- They can implement interfaces
- Can be passed as parameters
- They can swap out their instances for testing purposes
- Ability to be handled polymorphically. There may be an existence of multiple implementations.

This design pattern is most commonly implemented alongside a private constructor method, as well as a public static method in order to return the instance of the singleton. They are stored in a private static final variable (freecodecamp, 2023).

A singleton class should be used in order to guarantee that only a single instance of a class exists during the runtime of a project (stackoverflow, 2023).



(SpringHow, 2023)

In this diagram, the Instance is the singleton object. The object is defined with private, static and final keywords. Since this object should not be accessible outside, it should then be initialized at the class level and it should not be changed in any future instance (SpringHow, 2023).

The constructor is private in order to prevent other classes from using them to create an object of the Singleton class (SpringHow, 2023).

The `getInstance` method is a public method in order for it to be called by other classes. This method is static as well so that when it is called an object is not required by the other classes (SpringHow, 2023).

How to implement Singleton To A Project?

- Create a single instance of the Singleton inside of a class (SpringHow, 2023).
- Give a static method in order to return the reference of the instance (SpringHow, 2023)
- All the constructors should be declared as "Private" to ensure that any other component cannot create objects for that specific class (SpringHow, 2023).

Architectural Patterns VS Design Patterns

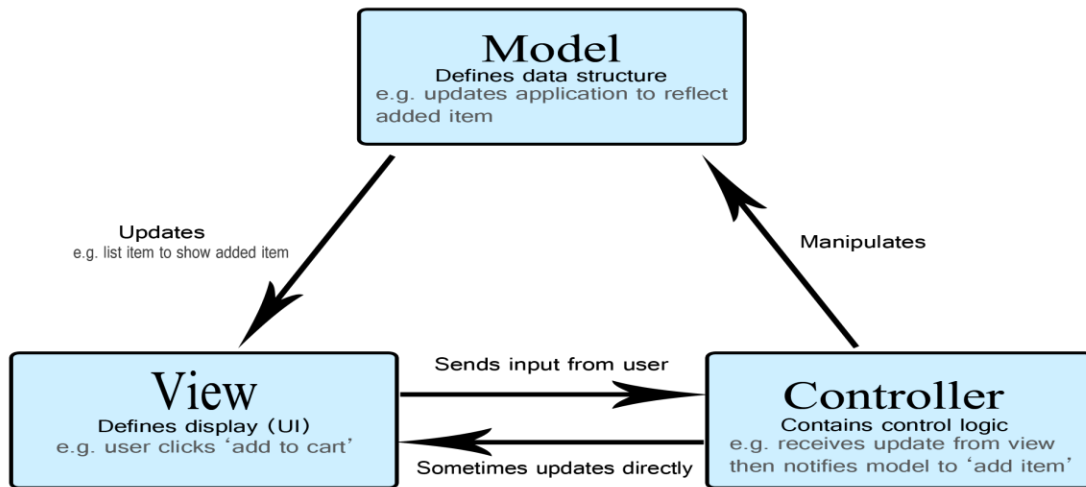
There is a very narrow difference between architectural and design patterns (Simform, 2023).

- Architectural patterns are the fundamental structural organisation for software systems, where as design patterns are a specification that may help in the implementation of a software (Simform, 2023).
- The role for architectural patterns is that they convert software characteristics into a high-level structure, whereas the role for design software creates a description of all the units of the software system in order to support the code (Simform, 2023).
- An example of architectural design is microservices, serverless and event-driven. Examples of design patterns are creational, structural, and behavioural (Simform, 2023).
- The problems that architectural patterns address is distributed functionality, protocols, system partitioning, interfaces, reliability, scalability, and security. The problems that design patterns address are the problems in software construction (Simform, 2023).

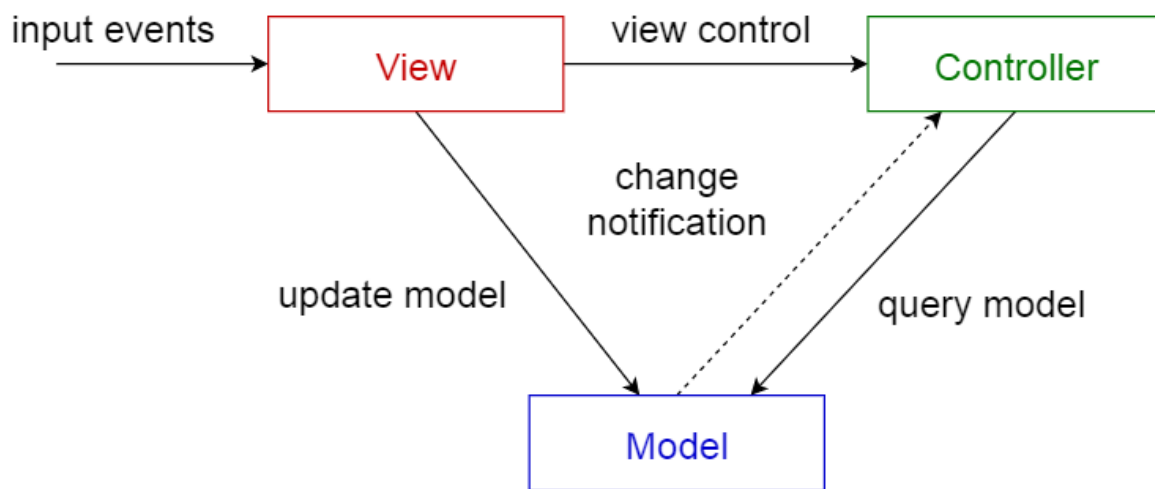
Architectural Pattern Of Choice And How It Will Be Used

The pattern that has been chosen to develop this website, is that of an Architectural Pattern. This pattern is known as MVC's (Microsoft, 2023).

MVC(Model-View-Controller)



This diagram shows the flow of how an MVC web application. (Foundation, 1998-2023)



(Medium, 2023)

This is a pattern which is commonly used to implement the user interfaces of a website, as well as its data and the controlling logic. MVC's emphasize the separation between the logic of the software's business and its display. The separation of this creates for a better division amongst the labour and improves the maintenance (Foundation, 1998-2023).

The 3 main components of MVC are the model, view, and controller.

- Models: This is the part of the application that implements the logic for the applications data domain. The model projects often retrieve and store the models state inside of a database (Foundation, 1998-2023) (Microsoft, 2023).
- View: These are the components which display the application to a user interface. Usually the user interface (UI) is created from the model data (Foundation, 1998-2023) (Microsoft, 2023).
- Controller: These are the components that handle the user interactions and select the views to render which display the UI. Inside of an MVC application, the view will only display the information, but the controller is what handles and responds to the input of a user and any other interaction (Foundation, 1998-2023) (Microsoft, 2023).

An MVC pattern aids in the applications creation which separate the different components of the application, while also providing a loose coupling between the elements (Foundation, 1998-2023) (Microsoft, 2023).

MVC's make the testing of applications easier than what it is to test a Web forms-based ASP.NET application (Foundation, 1998-2023) (Microsoft, 2023).

The coupling between the components of a MVC application promotes parallel development. This means that each developer can work independently from one another (Foundation, 1998-2023) (Microsoft, 2023).

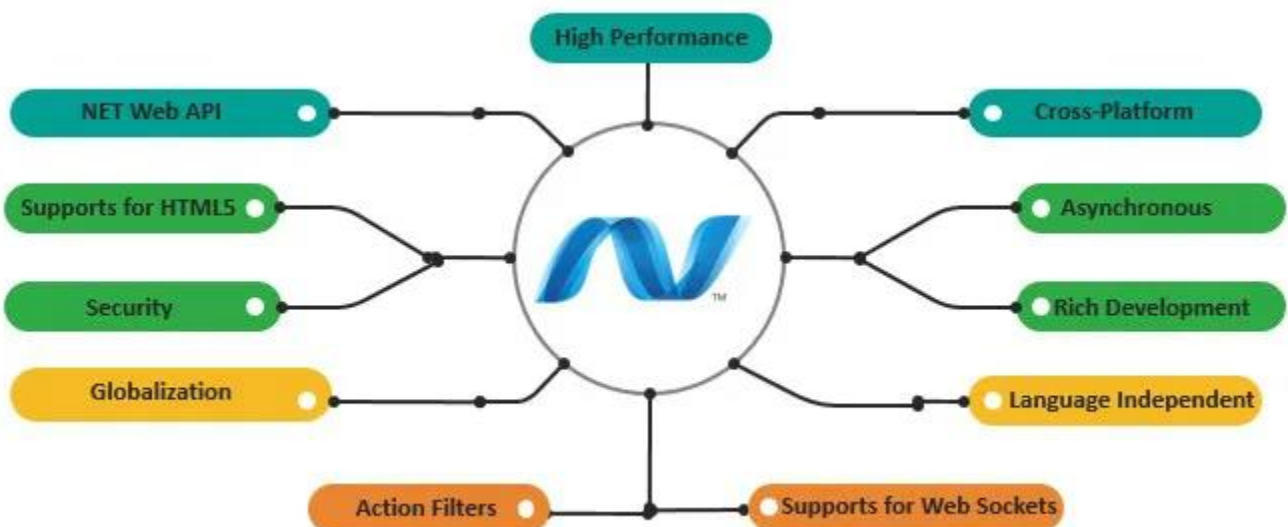
As well as being able to manage the complicity of a web site, the MVC pattern creates an application which is easier to test than that of a Web Forms-based ASP.NET application (Foundation, 1998-2023) (Microsoft, 2023).

Advantages Of Using MVC-Based Web Applications.

- It is easier to manage the complexity of a website by dividing an application into the core parts of MVC's (model, view, controller) (Microsoft, 2023).
- It does away with the view state as well as the server-based forms. This creates an ideal framework for the developers who want full control over the behaviour of the application (Microsoft, 2023).
- The use of a Front Controller pattern is used to process Web application requests through a single controller. Doing this then enables the developer to design the application which supports a rich routing infrastructure (Microsoft, 2023).
- MVC's provide a better support for test-driven development (Microsoft, 2023).
- MVC's and Web applications go hand in hand if the application is supported by a large team of developers and web designers who need a high degree of controller over the behaviour of the application (Microsoft, 2023).

Features of the ASP.NET MVC Framework

- This framework promotes the separation of the tasks, testability, as well as test-driven development in an application (Microsoft, 2023).
- This is an extensible and pluggable framework. The core components of MVC's are designed purely to ensure that they can become easily replaceable and or customised (Microsoft, 2023).
- MVC's contain a URL-mapping component which is an extremely powerful component which lets the user build web applications which contain comprehensible and searchable URLs (Microsoft, 2023).
- MVC's allows the user to use features such as a forms and Windows authentication, a URL authorisation, output a data caching, membership and roles, session, and the management of profile state, monitoring of health, configuration systems, and as well as the provider architecture (Microsoft, 2023).



www.educba.com

(Educba, 2023)

Password Hashing For Website

Salting a password means adding random values to a password prior to its hashing to improve the strength of it.

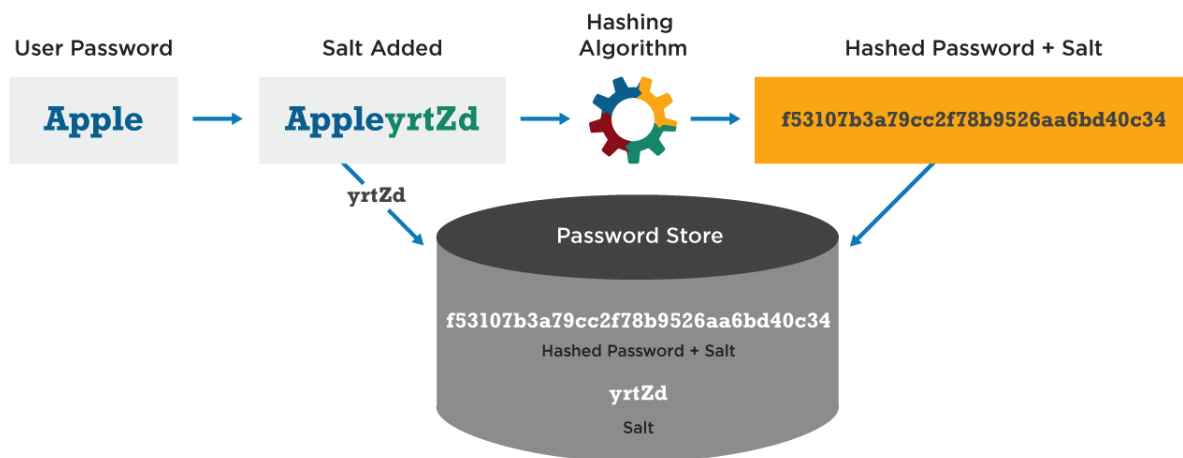
The MVC framework allows the user to add a CRUD aspect to the WEB Application. This aspect stands for Create, Read, Update, Delete.

With the use of CRUD, the website will be able to flow a lot smoother as the creating, deleting, and updating of farm stock; as mentioned by Farm Central; will be a lot easier to do and maintain (CrowdStrike, 2023).

The hashing of a password promotes safety and reduces the chance of an account being hacked. The password hashing technique that will be used is UTF-8 or SALT (PinkIdentity, 2023).

SALT password hashing adds random characters before and after a password prior to hashing in order to obfuscate the actual password (PinkIdentity, 2023).

Password Hash Salting



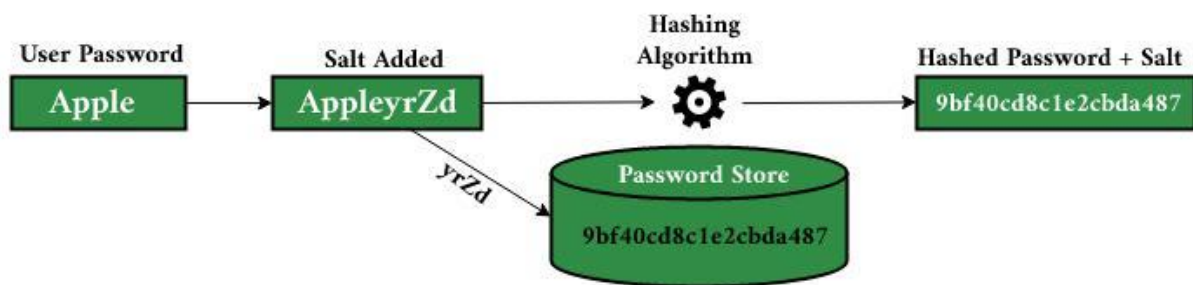
(CyberHoot, 2023)

This is a technique which is used to help protect a password that is stored in a database to prevent the reverse-engineering of it by hackers who might breach the environment (CyberHoot, 2023)

This method adds between 32 or more string characters to a password and then it hashes it. This is one of the most secure methods in order to protect a user's passwords that is stored for authentication in the future without the exposure of them in case the website is breached (CyberHoot, 2023). A downside of password salting is that the password will need to be hashed multiple times to increase the validity of its security.

To increase the validity of the password salting method, the user should also consider the following:

- Ensuring the salt string isn't too short and refrain from using the usernames as your salts (CyberHoot, 2023).
- The salt should be a string of 32 characters or longer in length (CyberHoot, 2023).
- The user should avoid using outdated hashing methods (MD5, SHA1) (CyberHoot, 2023).
- Hash the salted password multiple times (CyberHoot, 2023).
- Educate the employers/ workers on the site of the threats that they may face moving forward. Educate the user on the basics of cybersecurity that can either stop or prevent these attacks before they succeed (CyberHoot, 2023).



(GeeksForGeeks, What is Salted Password Hashing?, 2023)

The user on a website enters a password, when the password is submitted it is then salted (the password adds characters either to the beginning or to the end or both beginning and end of the password). The salted password is then hashed into a 32 character or longer string which is then pushed into a database for storage.

An Example of Hashing a Salted Password

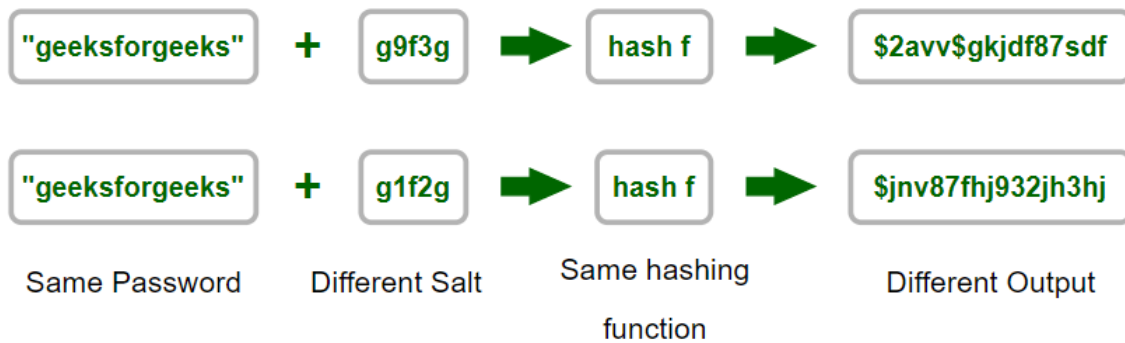
Password (Original Input):	h@ckPr00f
Salt (Unique Integer):	H13qY7YPCDJULY7I
Password + Salt Combo:	h@ckPr00fH13qY7YPCDJULY7I
Password Hash (Output):	\$argon2id\$v=19\$m=16,t=2,p=1\$SDEzcVk3WVBDREpVTFk3bA\$59cGxSHPjID4mrJYFxMFUQ

(Insights, 2023).

An example of password hashing using SALT (Insights, 2023).

In the first block, the user types out their desired password. The password is then hashed into the SALT format. The password and the SALT format are combined in the 3rd block to be hashed. The password is then hashed finally in the last block. The Password hash is generated to using the Argon2Id hashing algorithm (PinkIdentity, 2023).

The purpose of the multiple encryptions is to prevent, or make hacking of an account harder to do (PinkIdentity, 2023).

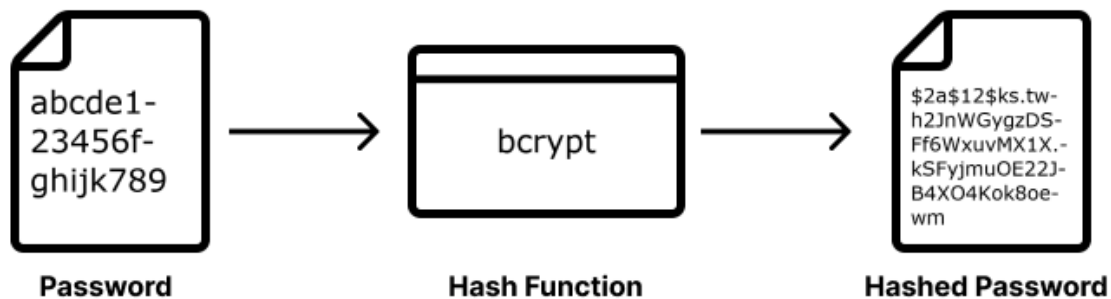


(GeeksForGeeks, What is salt in hashing?, 2023)

When the password is hashed using salt, it gives it its own unique string to prevent clashing with other passwords if they are the same. It hashes the passwords into its own unique identity (GeeksForGeeks, What is salt in hashing?, 2023).

The UTF-8 Password hashing method which is an encoding in bytes of Unicode code points to which each character takes one or more bytes (PinkIdentity, 2023).

Password Hashing



(Authgear, 2023)

References

- Authgear. (2023, April 16). *What Is Hashing?* Retrieved from <https://www.authgear.com/post/password-hashing-salting>
- betterProgramming. (2023, April 16). *What Is a Singleton?* Retrieved from <https://betterprogramming.pub/what-is-a-singleton-2dc38ca08e92>
- Cloudflare. (2023, April 16). *What is blob storage?* Retrieved from <https://www.cloudflare.com/learning/cloud/what-is-blob-storage/#:~:text=Blob%20storage%20is%20a%20type,storage%20areas%20called%20data%20lakes.>
- CrowdStrike. (2023, April 16). *WHAT IS CRUD?* Retrieved from <https://www.crowdstrike.com/cybersecurity-101/observability/crud/#:~:text=CRUD%20is%20the%20acronym%20for,in%20relational%20and%20NoSQL%20databases.>
- CyberHoot. (2023, April 16). *Password Salting*. Retrieved from <https://cyberhoot.com/cybrary/password-salting/>
- digitalOcean. (2023, April 16). *Singleton Design Pattern Best Practices with Examples*. Retrieved from <https://www.digitalocean.com/community/tutorials/java-singleton-design-pattern-best-practices-examples>
- Educba. (2023, April 16). Retrieved from <https://www.educba.com/features-of-asp-dot-net/>
- Foundation, M. (1998-2023, April 16). *MDN Web Docs*. Retrieved from <https://developer.mozilla.org/en-US/docs/Glossary/MVC>
- Foundation, M. (2023, April 16). *CSS: Cascading Style Sheets*. Retrieved from <https://developer.mozilla.org/en-US/docs/Web/CSS>
- Foundation, M. (2023, April 16). *The "why" of web performance*. Retrieved from https://developer.mozilla.org/en-US/docs/Learn/Performance/why_web_performance
- freecodecamp. (2023, April 16). *The Singleton Design Pattern*. Retrieved from <https://www.freecodecamp.org/news/4-design-patterns-to-use-in-web-development/#:~:text=The%20Singleton%20Design%20Pattern&text=That%20prevents%20multiple%20instances%20from,global%20state%20of%20an%20application.>
- GeeksForGeeks. (2022, August 25). *Dynamic Websites*. Retrieved from <https://www.geeksforgeeks.org/dynamic-websites/>
- GeeksForGeeks. (2023, April 16). *What is salt in hashing?* Retrieved from <https://www.geeksforgeeks.org/how-to-hash-passwords-in-python/>
- GeeksForGeeks. (2023, April 16). *What is Salted Password Hashing?* Retrieved from <https://www.geeksforgeeks.org/what-is-salted-password-hashing/>

- Hat, R. (2023, April 16). *14 software architecture design patterns to know*. Retrieved from <https://www.redhat.com/architect/14-software-architecture-patterns#:~:text=There%20are%20numerous%20advantages%20to,patterns%20you%20can%20tap%20into.>
- Insights, I. (2023, April 16). *Why Using a Password Salt and Hash Makes for Better Security*. Retrieved from <https://sectigostore.com/blog/why-using-a-password-salt-and-hash-makes-for-better-security/>
- javarevisited. (2023, April 16). *Difference between UTF-8, UTF-16 and UTF-32 Character Encoding?* Retrieved from <https://javarevisited.blogspot.com/2015/02/difference-between-utf-8-utf-16-and-utf.html>
- Medium. (2023, April 16). *10 Common Software Architectural Patterns in a nutshell*. Retrieved from <https://towardsdatascience.com/10-common-software-architectural-patterns-in-a-nutshell-a0b47a1e9013>
- Microsoft. (2023, April 16). *ASP.NET MVC Overview*. Retrieved from <https://learn.microsoft.com/en-us/aspnet/mvc/overview/older-versions-1/overview/asp-net-mvc-overview>
- modernescpp. (2022, September 25). *The Singleton: Pros and Cons*. Retrieved from <https://www.modernescpp.com/index.php singleton-pros-and-cons#:~:text=A%20Singleton%20is%20a%20global,a%20means%20to%20protect%20globals.>
- PinkIdentity. (2023, April 16). *Encryption vs. Hashing vs. Salting - What's the Difference?* Retrieved from <https://www.pingidentity.com/en/resources/blog/post/encryption-vs-hashing-vs-salting.html#:~:text=Password%20salting%20adds%20random%20characters,of%20passwords%20won't%20work.>
- refactoring.guru. (2023, April 16). *Singleton*. Retrieved from <https://refactoring.guru/design-patterns/singleton#:~:text=Use%20the%20Singleton%20pattern%20when,for%20the%20special%20creation%20method.>
- Sectigo. (2023, April 16). *4 Reasons Why Website Security Is Important*. Retrieved from <https://sectigo.com/resource-library/why-is-website-security-important-and-why-should-businesses-care>
- Simform. (2023, April 16). Retrieved from <https://www.simform.com/blog/software-architecture-patterns/#:~:text=Famous%20examples%20of%20architectural%20patterns,and%20presentation%20abstraction%20control.>
- SpringHow. (2023, April 16). *Singleton Design Pattern*. Retrieved from <https://springhow.com/singleton-pattern/>
- StackExchange. (2023, April 16). *Implications of hashing to UTF-8 in a cryptographic function*. Retrieved from <https://security.stackexchange.com/questions/98480/implications-of-hashing-to-utf-8-in-a-cryptographic-function#:~:text=UTF%208%20is%20an%20encoding,that%20is%20required%20at%20all.>

stackoverflow. (2023, April 16). *On design Patterns: When Should I Use The Singleton?* Retrieved from <https://stackoverflow.com/questions/228164/on-design-patterns-when-should-i-use-the-singleton#:~:text=A%20singleton%20should%20be%20used,of%20pattern%20can%20be%20vital.>

tutorialsPoint. (2023, April 16). *Design Pattern - Singleton Pattern*. Retrieved from https://www.tutorialspoint.com/design_pattern/singleton_pattern.htm#:~:text=This%20pattern%20involves%20a%20single,the%20object%20of%20the%20class.

UpGuard. (2023, April 16). *Website Security Checklist: Protect Your Website in 2023*. Retrieved from <https://www.upguard.com/blog/the-website-security-checklist>