



# PROJECT REPORT

***Revised document with changes***

Prepared For:

**Farm Central**

By Revali Reddy.st10084347.BCA3.Group2

123 Ocean Rd  
Durban, South Africa

[www.itsolutions.com](http://www.itsolutions.com)

# Contents Page

<b>Introduction</b>	03
<b>Non-functional Requirements</b>	
Performance	04
Usability	05
Visual Appeal	06
Security	08
Reliability	09
Data Accuracy	11
Training	12
<b>Design Patterns</b>	
Observer pattern	13
Factory Pattern	15
Singleton Pattern	16
<b>Architecture Patterns</b>	
RESTful architecture pattern	19
Model-View-Controller architecture	21
Event-driven architecture	22
<b>Conclusion</b>	24

# Introduction

The content of this report aims to explain the successful development of a solid, user-friendly platform that will improve Farm Central's stock management system , boost productivity and ease the production processes of Farm Central. Farm central sells different types of wood . Hence , a stock management system that aims to track stock that is going out and coming in as well as which farmer each item belongs to is needed to store information about this retail wood business. These items may include timber , oak , pine etc. This report pays special attention to financial management , store management, marketing and end-user requirements. Hence, specific non-functional requirements have been identified and addressed as well as the impact these have on the software development plan. Specific design and architecture patterns have also been highlighted to provide readers of this report with a better understanding of the most ideal stock management website structure.



Image: Wood/Bailey, Ben "How to Start a Firewood Business in 5 Steps" Ventures (8 Oct 2020), [ventures.com/how-to-start-a-firewood-business-5-steps/](https://ventures.com/how-to-start-a-firewood-business-5-steps/) Accessed 16 Apr 2023



# Non-Functional Requirements



1. The following non-functional requirements are of high-importance to the Head of Marketing.

## 1.1

### Performance

The system should be able to handle a huge volume of data and transactions without degrading performance. The website should be provide immediate responses, with quick load times and as little downtime as possible. It is vital to enable for continuous stock level updates. Hence, data must be processed and displayed in real-time (*White.D. ,2019*).

#### Addressing the requirement:

- **Determine the precise performance needs** such as reaction time, data loading speed, scalability, and dependability ,e.g. An accountant will be able to make judgments about ordering and stocking products because the website must accurately manage inventory levels in real-time.
- **Establish performance measures** including page load time, server response time, database query performance, and error rate by determining how effectively the website will satisfy those needs, e.g. The store manager can be assured that customers can browse and shop on a website quickly and simply if the loading speed is swift.
- **Evaluate performance** by finding any potential bottlenecks or opportunities for improvement ,e.g. The head of marketing can assured that customers have uninterrupted access to the website and can make purchases there.
- **Execute performance optimization** by optimizing database queries, caching data, reducing picture sizes, and improving server setups to improve performance, e.g. A high level of customer engagement can lead to increased sales and customer loyalty.
- **Employ monitoring tools** to identify and resolve any problems that may occur ,e.g. This can benefit customers by providing them with a good user experience.
- **Make any necessary changes** to the website's performance requirements based on feedback from users such as the accountant , head of marketing and store manager.
- **Conduct performance testing** to identify performance bottlenecks to make sure it complies with the necessary performance standards .

(*AltexSoft ,2019*)

## Impact on the software development plan

- The website's software architecture design must take into account the infrastructure, frameworks, and technologies that can support the desired performance. **Design decisions** made during the development process might also **influence performance needs** e.g. If the software is required to handle big volumes of data or heavy processing, it may necessitate a different architecture or design approach than if it is simply required to handle modest amounts of data.
- The software development plan must include preparations for the website using a distributed architecture, caching, and load balancing that can scale to **accommodate a large number of users**.
- The development plan must guarantee **effective security, dependability, and correctness** of financial data.
- Programmers will need to **optimize code** by making algorithms more efficient, fewer database queries and transferring less data.
- Achieving performance requirements may necessitate the adoption of performance-optimized technology or tools involves **more study and review of other possibilities**, which **may impair the development timeline**.
- Performance testing should be part of the software development plan to make sure it complies with the necessary performance standards. This may necessitate **the addition of testing resources, technologies, and infrastructure**.
- The project's **budget** may need to **allocate additional resources** to ensure that the performance standards can be met.

(Hamilton, T. 2019).

### 1.2

## Usability

Users should be able to simply explore and operate the system thanks to its user-friendly interface. e.g. The system must make sure there are no delays or confusion so the farmers can promptly and easily deliver their goods (Honig, J. 2022.).

### Addressing the requirement:

- **Engage with user research** to better identify the requirements and expectations of each user of the website e.g. user research may include utilizing surveys, interviews, or usability testing
- **Apply natural design concepts** to make the website simple to use and navigate. e.g. The store manager can be assured that the inventory management capabilities are simple to use, with brief item labels and easy-to-navigate inventory levels.
- **Inform people** with language that is simple and succinct.
- **Make it simple for consumers** to enter information by using user-friendly forms.
- **Offer assistance and support items** like FAQs and user manuals to assist users in completing tasks on the website. e.g. Users will be educated on how to manage their profiles, order histories, and automated user communication.
- **Perform usability testing** on actual users to determine flaws with the functionality or aesthetics of the website and to verify that the system satisfies the needs and expectations of the stakeholders. e.g. This assures the head of marketing that the target audience's wants and expectations are met , which improves the customer experience and promotes engagement and loyalty.
- **Ensure that the website is usable and accessible** on a range of gadgets, such as desktop computers and mobile devices by using responsive design e.g. The accountant can be guaranteed that the functions of the website are responsive for mobile devices, with clean interfaces and little extraneous stuff.

(AltexSoft ,2019).



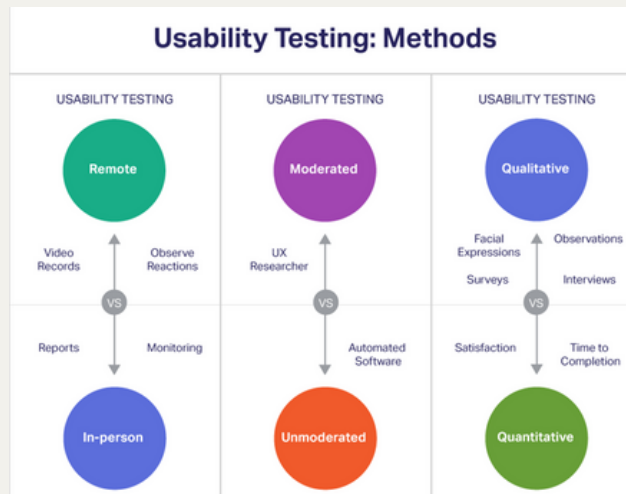


Image: Usability Testing / Stefan Ivanov UX Architect et al. (no date) What is usability testing (2021 guide). Infragistics blog. Infragistics Community. Available at <https://www.infragistics.com/community/blogs/by/infragistics/posts/what-is-usability-testing> (Accessed April 13, 2023).

### Impact on the software development plan:

- The software development plan must be **developed according to the target audience**, their needs, and their behaviors.
- The development plan must be **designed through iterative design, testing, and evaluation** to ensure that the software meets the user's needs.
- Developers must **follow design standards and guidelines** such as typography, color schemes, layout, and navigation to ensure that the software is consistent and familiar to users.
- Developers must **test the software** with real users through usability testing, user acceptance testing, and accessibility testing to identify usability issues and validate the software's usability.
- Developers must **provide clear documentation and training** through include user manuals, video tutorials, and online help to aid users in understanding how to use the software effectively.
- The development plan must **set aside enough funding** to satisfy usability standards.

(Cudd. G., 2023).

## 1.3

### Visual Appeal

For the system to leave a good impression on the farmers and inspire confidence in it, it should be visually appealing and have a contemporary feel (*Interaction Design Foundation*, 2019).

#### Addressing the requirement:

- **Create a visual aesthetic** that reflects the brand and the needs of the target market in terms of color schemes, typography, and layout. e.g. The head of marketing can assess the product portfolio more easily with the help of a visually appealing stock management system that offers an understandable visual depiction of the products.
- **Utilize high-quality graphics and photos** to allow to the website to be more visually appealing to users. e.g. Users are able to make rational choices regarding their purchases with a system that displays product information in an intuitive manner.
- **Inspect the uniformity** of the website's visual style. e.g. Accountants can save time by having immediate access to essential data and minimizing the time it takes to obtain the information they require using a visually appealing stock management system.





- **Incorporate white spaces into the websites design** to ensure that websites layout is tidy and well-organized. This can make content easier to read and improve the aesthetic appeal of a website. e.g. A tidy system may help the business execute orders precisely and immediately. This decreases the possibility of order errors or delays , ultimately allowing the business to provide superior customer services.
- **Employ a visual hierarchy** to emphasize key information. e.g. This may allow the store manager to immediately detect and fix problems like low stock levels or slow-moving merchandise.
- **Consider a responsive design** to ensure that the website can be used on multiple devices such as PC's ,tablets, and mobile phones.
- **Carry out usability testing** to detect faults with the website's aesthetic design. This must be done with actual users.

(McConnell. M,2018,  
AltexSoft,2019).

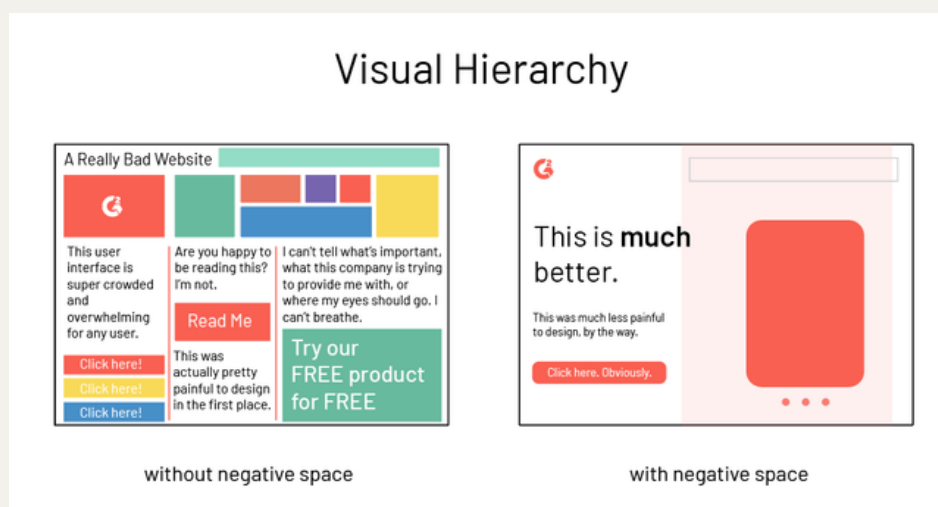


Image: Visual Hierarchy | Bretous, M. (2022) A non-designer's guide to visual hierarchy [best practices + examples], HubSpot Blog, HubSpot. Available at: <https://blog.hubspot.com/marketing/visual-hierarchy> (Accessed: April 13, 2023).

### Impact on the software development plan:

- The software development plan must include a **time- and resource-intensive UI design** that may **require extra time and expenses** that need to be taken into account.
- The development plan must include a **well-designed user interface** that may can **boost the product's usability** and lead to more usage and user engagement.
- The software development plan may need **developers and designers to collaborate and communicate** which can **delay the development process**.
- A software development plan that aims to achieve a **visually appealing interface must undergo extra testing** to ensure that it functions well and adheres to the design specifications. This can necessitate **more testing time and materials**.
- The development plan must **guarantee uniformity and branding** throughout the product development process to provide a unified and professional image.

(Interaction Design Foundation , 2019).

## 2. The following non-functional requirements are of high-importance to the Accountant.

### 2.1

## Security

In order to protect sensitive data, such as financial information, the system must comply to security standards and regulations and have strong security features, like user authentication and access control features (Ed.gov. 2020).

### Addressing the requirement:

- **Backup data** on a regular basis to guarantee that it can be restored in the event of data loss or system failure. To safeguard critical information such as client data, inventory, and financial information, encrypt data using SSL/TLS certificates ,e.g. Users may get a copy of their data that contains information about the stocks they have and the numbers they have .
- **Do regular penetration testing**, vulnerability scanning, and security audits to monitor and test the website's security.
- **Perform risk assessments** on the website's infrastructure, data storage, access controls, and user authentication processes to identify and mitigate any security risks and vulnerabilities ,e.g. Any performance problems may affect the timeliness and accuracy of financial reporting that may effect the accountant.
- **Establish effective security measures** such as firewalls, SSL encryption, multi-factor authentication, and access controls, as well as ensure the website is up to date and free of vulnerabilities to safeguard the website and its users ,e.g. The store manager may require that access to specific portions of the stock management website be restricted.
- **Adhere to data privacy standards** such as GDPR or CCPA, as well as financial regulations such as PCI-DSS. e.g. This may ensure the head of marketing that personal data gathered is handled safely
- **Offer security best practices** training to all staff members ,e.g. General staff can feel familiar with the website through workshops that educate them on things such as setting strong passwords, preventing phishing scams, and recognizing potential security threats.

(Lipan. M,2021).

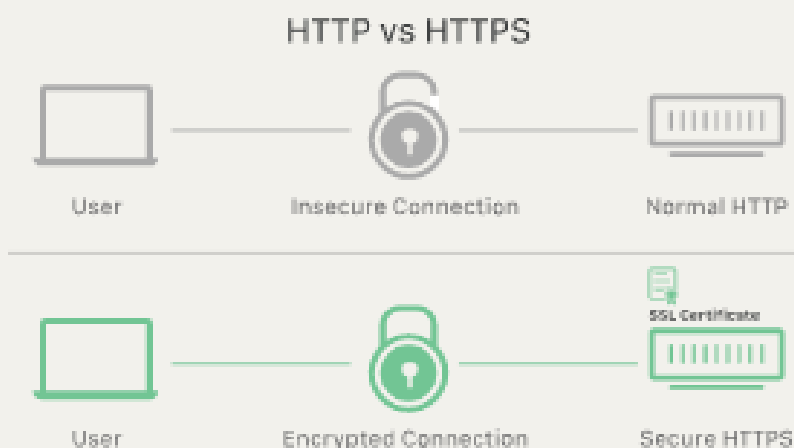


Image: Process Validation | Pharmaguideline (no date) Validation program in Pharmaceutical Industries, Pharmaguideline. Available at: <https://www.pharmaguideline.com/2017/09/validation-program-in-pharmaceutical.html> (Accessed: April 13, 2023).

### Impact on the software development plan:

- The development plan should involve **using secure coding practices** such as input validation, data sanitization, and using secure authentication methods.
- Developers must conduct **regular security audits** that cover network, server, and application layers to identify potential vulnerabilities or weaknesses in the website's security infrastructure.
- Developers must **monitor the website** by maintaining logs for auditing purposes as well as continuously checking the website for abnormal activities.



- To ensure that the website can recover rapidly from any security incidents and reduce the impact on business operations ,**a disaster recovery and business continuity plan must be drawn.**
- Developers may need to **modify the development methodology** to guarantee the security of the software e.g. The website will need to a security concepts and practices .
- The choices taken in terms of design during the development phase may include **needing a specific architecture or design approach** if it needs to be encrypted or authenticated.
- The software development plan must **conduct additional analysis, testing, and documentation**, which could **delay the development process** to ensure security compliance.

(Harvey.S, 2020).



Image:SecurityArchitecture,Locketal.(2022)LearnthedifferencesbetweenHTTPandHTTPSprotocols,DirectLineDevelopment,Availableat<https://directlinedev.com/blog/difference-between-http-and-https/>(AccessedApril13,2023).

## 2.2

## Reliability

The website must operate reliably and consistently without interruption, mistake, or failure. For accountants who need exact financial data, this is crucial (Anon, 2021).

### Addressing the requirement:

- **Establish reliability measures** such as uptime, mean time to failure, mean time to repair, and error rates, e.g. Accountants can develop standardized processes for logging transactions or balancing accounts to ensure reliability.
- **Find potential areas for reliability improvement** , e.g. Automating repetitive processes carried out by accountants can contribute to greater reliability by lowering the possibility of data entry and record-keeping mistakes.
- **Utilize redundancy techniques** to ensure that the website is accessible even in the event of a breakdown e.g. Developers can deploying numerous servers, load balancing, and cloud-based hosting services.
- **Adopt a regular backup system** to prevent data loss in the event of theft ,natural catastrophes, or other unforeseen difficulties e.g. The store manager can instantly restore data from a backup, minimizing downtime and guaranteeing that business can continue as usual.
- **Employ monitoring tools** to find any problems that can affect the dependability of the website, such as server faults, database difficulties, and other performance problems e.g. The head of marketing can receive real-time notifications, creating awareness of problems or concerns with the website right away. They can quickly solve problems as they arise and lessen their negative effects on the customer experience.
- **Perform routine testing** to maintain the website's dependability in situations like excessive traffic or system breakdowns e.g. This ensures the head of marketing that the website is running as efficiently as possible

(www.javatpoint.com ,2011).

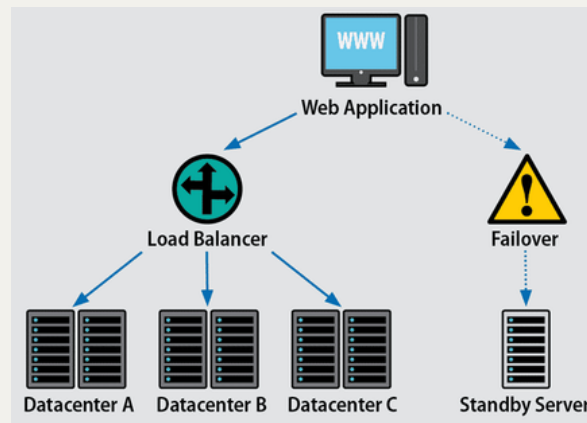


Image: Redundancy Techniques | Davies, N. (2021) What is RAID 10 - how raid 10 works, Enterprise Storage Forum. Available at: <https://www.enterprisestorageforum.com/software/what-is-raid-10-how-raid-10-works/> (Accessed April 13, 2023).

### Impact on the software development plan:

- The development plan must account for the ability to **recognize and repair mistakes and failures**, such as improper inputs, unexpected user behavior, or hardware failure. Hence , the software development plan will need to include testing tools , infrastructure and resources.
- Developers must ensure that the program must **undergo extensive testing** by performing functional testing, integration testing, performance testing, and stress testing to emulate real-world settings and find reliability concerns.
- Appropriate **error handling procedures must be implemented** in the development plan to identify and elegantly manage errors. This entails more testing, debugging, and quality assurance work, which can take a long time and be expensive to produce.
- To identify faults and take timely corrective action, **the software's performance must be tracked** using log analysis, system performance monitoring, and user input. As a result the development plans time and cost are affected because of additional quality assurance, testing, and bug-fixing activities.
- To assist users in understanding the software's dependability and how to resolve problems ,**the software's behavior, intended results, and potential failure situations must be described** in the development plan to guarantee the program's long-term dependability.

(Creately Blog. 2011 , ReQtest ,2020).

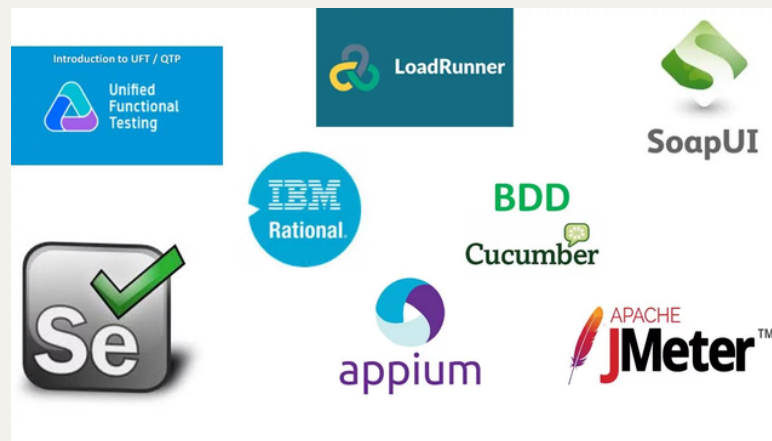


Image: Testing Tools | Reddy, G.C. (2021) Software test tools quiz, Software Testing. Available at: <https://www.gcreddy.com/2021/10/software-test-tools-quiz.html> (Accessed April 13, 2023).

## Data Accuracy

To ensure that each farmer's income is accurately documented and that there are no anomalies in the financial records, the system should assure accurate and up-to-date data regarding stock levels, incoming and exiting stock, and whose farmer each item belongs to (*Solutions.M,2022*).

### Addressing the requirement:

- **Determine particular requirements for data accuracy**, such as correct inventory figures for financial reporting needs or accurate stock levels for marketing campaigns. *e.g.* An accountant can avoid overvaluing or undervaluing inventory, which can have substantial financial ramifications, by having the right data.
- **Create a data governance plan** that guarantees that data is correct, comprehensive, and consistent across the stock management website. This includes roles and responsibilities for controlling data quality as well as policies and procedures for data entry, storage, and validation. *e.g.* Data governance allows the head of marketing to maximize marketing initiatives to swiftly modify promotional activities if stock of a product is running low.
- **Utilize automated technologies** to verify data accuracy at the moment of data entry. This includes data validation rules and automatic error alarms that indicate wrong or incomplete data. *e.g.* This aids a store manager in having real-time information on inventory levels and prevents overstocking or understocking of inventory as well as detecting missing inventory.
- **Analyze and monitor data frequently** to spot problems with its accuracy, such as missing or wrong data.
- **Provide instructions to users** on the value of precise data entry as well as how to do so.
- **Clearly state who is in charge** of keeping data accurate, and give them the resources and equipment they need, such as a data quality team or data stewards. *e.g.* This ultimately improves user support since the support staff can promptly look up inventory levels and give precise instructions on how to fix the problem when a customer has a problem with a product.

(*AltexSoft,2019*).

### Impact on the software development plan:

- To guarantee that all data entered into the system is correct, full, and consistent, **set data quality criteria**. The database schema, data models, and data storage techniques must all be optimized.
- Make sure that the data entered into the system is accurate and complies with set criteria by **using data validation techniques** to validate data kinds, ranges, and forms.
- **Include error handling** in the development plan to make sure that any data discrepancies or errors are quickly found and fixed.
- To ensure that only authorized users may enter data into the system, the development plan must **utilize data entry restrictions** such password-protected accounts and role-based access controls.
- **Utilize automated data cleansing technologies** to find and fix data flaws and inconsistencies during the development process.
- The software development plan must include **routine data audits** to make sure the data is accurate and current by checking it for completeness, accuracy, and consistency.
- The software development plan must **allocate enough money and time** for additional of resources like testing tools and staff to meet the demands for data accuracy.

(*Shen.S,2019*).

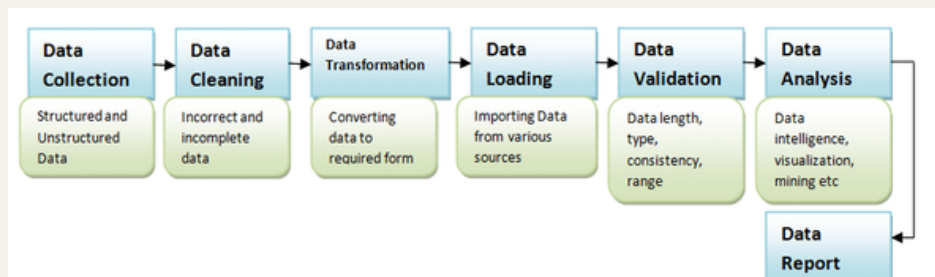


Image :Data Validation techniques (Kislaya Chakrabarti (2021)Improving data validation using machine learning, a new way of seeing big data| Available at: <https://www.researchgate.net/profile/Kislaya-Chakrabarti> (Accessed: April 13, 2023).

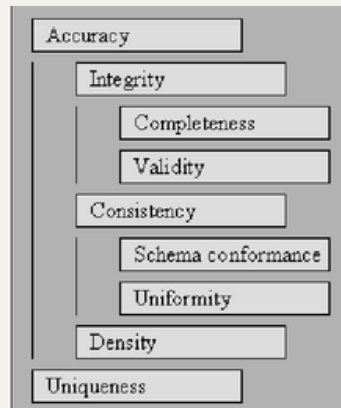


Image :Hierarchy Data quality criteria | Problems, methods, and challenges in comprehensive data ... - researchgate (no date). Available at: [https://www.researchgate.net/profile/Heiko-Mueller/publication/228929938\\_Problems\\_methods\\_and\\_challenges\\_in\\_comprehensive\\_data\\_cleansing/links/09e415101b58541e2c000000/Problems-methods-and-challenges-in-comprehensive-data-cleansing.pdf](https://www.researchgate.net/profile/Heiko-Mueller/publication/228929938_Problems_methods_and_challenges_in_comprehensive_data_cleansing/links/09e415101b58541e2c000000/Problems-methods-and-challenges-in-comprehensive-data-cleansing.pdf) (Accessed: April 13, 2023).

### 3. The following non-functional requirements are of high-importance to the Store Manager.

## 3.1 Training

To guarantee that training can be delivered effectively to train shop workers to utilize the system, the system should be simple to learn and train the staff to use (AltexSoft,2019).

#### Addressing the requirement:

- **Determine the user's skill gaps** and necessary development. *e.g.* This assures a store manager that they will be taught how to successfully handle sales and purchases and promptly determine when stock needs to be refilled with a greater grasp of the system.
- **Create user training resources**, such as step-by-step instructions, video tutorials, and user manuals. *e.g.* An accountant can reconcile accounts more quickly by receiving training in stock management systems as they can easily spot inconsistencies and promptly address them if they are aware of how inventory is monitored and handled.
- **Offer interactive training** that engages users and enables them to practice using the stock management website, such as live training sessions, webinars, and online courses. *e.g.* End consumers can comprehend product availability and stock levels better. This can assist the business in providing clients with better product recommendations and enhancing their overall purchasing experience which in turn increases revenue.
- **Grant users access** to online documentation, user forums, and help desk support to allow them to continue to make efficient use of the stock management website.
- **Provide users with refresher classes** to assist them keep up their knowledge and skills. *e.g.* The head of marketing can more effectively control supply levels and time promotions and discounts by having a deeper understanding of the system.
- **Utilize information obtained** from user surveys to make the training process and the training materials better.

(www.linkedin.com,(n.d)).

#### Impact on the software development plan:

- The software development plan must include a **time- and resource-intensive UI design** that may **require extra time and expenses** that need to be taken into account.
- The development plan must include a **well-designed user interface** that may can **boost the product's usability** and lead to more usage and user engagement.
- The software development plan may need **developers and designers to collaborate and communicate** which can **delay the development process**.
- A software development plan that aims to achieve a **visually appealing interface must undergo extra testing** to ensure that it functions well and adheres to the design specifications. This can necessitate **more testing time and materials**.
- The development plan must **guarantee uniformity and branding** throughout the product development process to provide a unified and professional image.

(Insights.M,2018).

# Design Patterns

---



## 4.1

### Observer Pattern

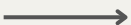
This pattern is useful when there is an interest in an object's status and notifications are desired whenever it changes. A subject monitors the conditions of another object whilst the observer keeps track. Event listeners for graphical user interfaces (GUI) applications frequently employ the observer pattern ([www.digitalocean.com](http://www.digitalocean.com) , (n.d.).

#### Structure

The principle of loose coupling between items that interact with one another is supported. This enables the effective data flow to other objects without altering the Subject or Observer classes. Observers may be added or removed at any time. A one-to-many dependency is created by the action of 'whenever one object changes its dependents are automatically notified and updated'. Hence the subject can be contact by any Observer whenever necessary ([www.tutorialspoint.com](http://www.tutorialspoint.com), (n.d.).

#### Relevance

- **Accurate records can be maintained and mistakes can be prevented** using the observer pattern. This is a result of the observer pattern allowing an accountant , for example , to monitor fluctuations in real-time stock levels to ensure the value of inventory on hand is appropriately reflected in the company's financial statements. A financial transaction module can alert any observers when transactions like inventory modifications, returns or refunds happen.
- **Rational judgements about promotions and advertising can be made** using the observer pattern. The marketing module can alert all observers, including the head of marketing, when marketing initiatives like discounts , sales or promotions, are introduced. Ultimately ,particular sales of products can be tracked to recognize patterns so that marketing plans can be modified accordingly. This is a result of the observer pattern being able to notify the head of marketing when these initiative are launched.



- **Wise decisions regarding reordering and restocking** can be taken with the integration of the observer pattern. This allows a store manager to ensure that extra inventory is not taking up space that can be utilized in another way and that popular items are always in stock because the observer pattern may be able to send alerts for occurrences like stock transfers, low inventory levels and stock modifications. By utilizing a stock management module, the store manager can be alerted when the above occur.
- **The processing load on the server can be decreased** with use of the observer pattern. This occurs when components can be updated without all other components being updated each time there is a change in stock levels of prices

(Caballero.C,2021).

### How will this design pattern be applied to the project?

- **A class must be created for the subject**. The methods that will be utilized by the observers to receive notifications should be defined by this interface or abstract class e.g. Stock is the subject. An alert may be sent as a result of fluctuations in stock.
- **An interface for the observer must be created**. The methods that will be implemented by the observer objects to receive notifications from the subject should be defined by this interface or abstract class e.g. Stock managers are observers. They may want to monitor these stock fluctuations.
- **The subject class should keep track of all observers** and include methods for adding, removing, and alerting observers to state changes.
- **The observer class must define the actions to be executed** when they receive notifications from the subject e.g. All stock managers(observers) will be alerted when alterations in stock items(subject) occur. This is referred to as the update method.
- **Procedures for the observers to sign up for alerts must be available** in the subject class. e.g. Stock managers(Observers) must ensure that they are registered to be informed when stock(subject) changes.
- **When the subject changes, alert the observer's**. e.g. When Stock managers alerted, they must respond to the changes of the stock(subject) as well update the website for users to view the changes in stock(subject).

(Caballero.C,2021,

[www.tutorialspoint.com](http://www.tutorialspoint.com). (n.d)).

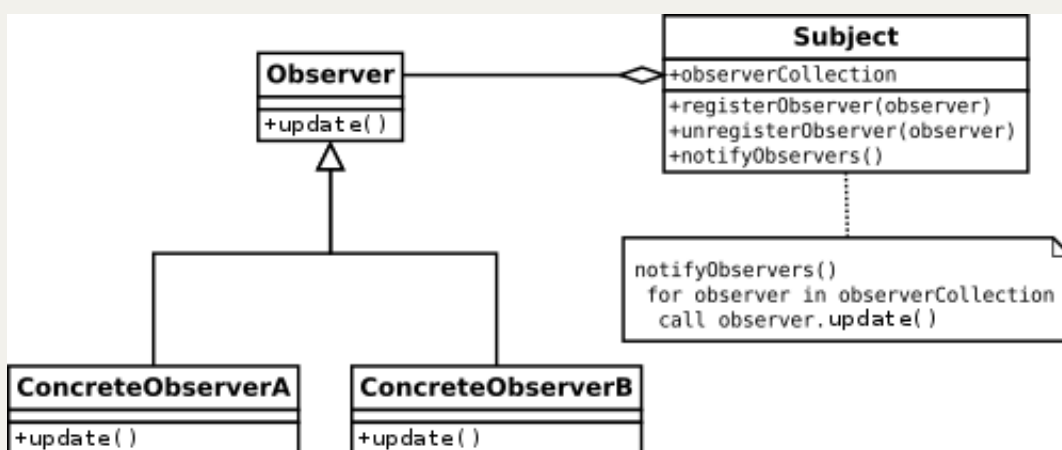


Image Diagram of Observer design pattern GeeksforGeeks(2016,ObserverPattern/Set1(Introduction)[online]Available at <https://www.geeksforgeeks.org/observer-pattern-set-1-introduction/> [Accessed 12Apr2023]



## Factory Pattern

The factory pattern includes objects that are built without disclosing the creation logic to the client. This pattern gives subclasses the ability to change the type of objects that will be created whilst utilizing a standard interface for producing objects in a superclass. The factory pattern enables the system to be developed in a more resilient, less connected, and simple to extend way (*www.tutorialspoint.com. (n.d.)*).

### Structure

Subclasses are given the option of selecting the kind of objects to produce. The code will only interact with the resulting interface or abstract class, allowing it to function with any classes that extends or implement that interface. This is a result of removing the requirement to tie application-specific classes into the code. Hence, the factory pattern encourages loose coupling. This fits the SOLID principles. (*Refactoring Guru, 2014*).

### Relevance

- **The report creation logic can be divided up into different factories** using the Factory pattern. The accountant can request a precise report produced by factories. This aids the accountant as a number of reports, including sales, inventory and financial must be produced.
- **Create several lists of products that are required by the marketing director.** This list includes highest-rated, best-selling and on-sale items. The logic for creating product lists can be divided into different factories by utilizing the Factory design. A precise report can be produced by each factory for the head of marketing. This is relevant for the flexibility requirement of the website.
- **The inventory management logic can be divided into different factories** using the Factory design. The factories can then develop the precise inventory management plan that the store manager requests. This aids the store manager since inventory such as durable, seasonal, or consumable goods need to be managed as well as require expandability.
- **A class may be unsure about the subclasses** it will need to build but this class may require that the objects to be produced be specified by its subclasses. e.g. stock items may have many different categories that may currently unknown or not yet determined. The factory design pattern allows the parent classes decide to create objects for their child classes, allows for good testability and significant method names.

(*Stack Overflow. (n.d.), IONOS Digital Guide. (n.d.)*).

### How will this design pattern be applied to the project?

- **A superclass for various types of objects can be produced from an abstract class** that outlines the procedures for producing stock management-related objects must be created. e.g. This may include a "stock" class.
- **A distinct kind of constructible object is represented** in each subclass that carry out the abstract class methods. e.g. This can include "tangible goods" and "non-tangible goods".
- **Objects for the factory class should be created** depending on user preferences, roles or requests from the abstract class.
- **The website will receive a request from a user** to produce the appropriate object based on the user's role. This will be invoked by the factory class. The website will carry out the desired operation based on the created object.

(*Csulb.edu, 2023*).



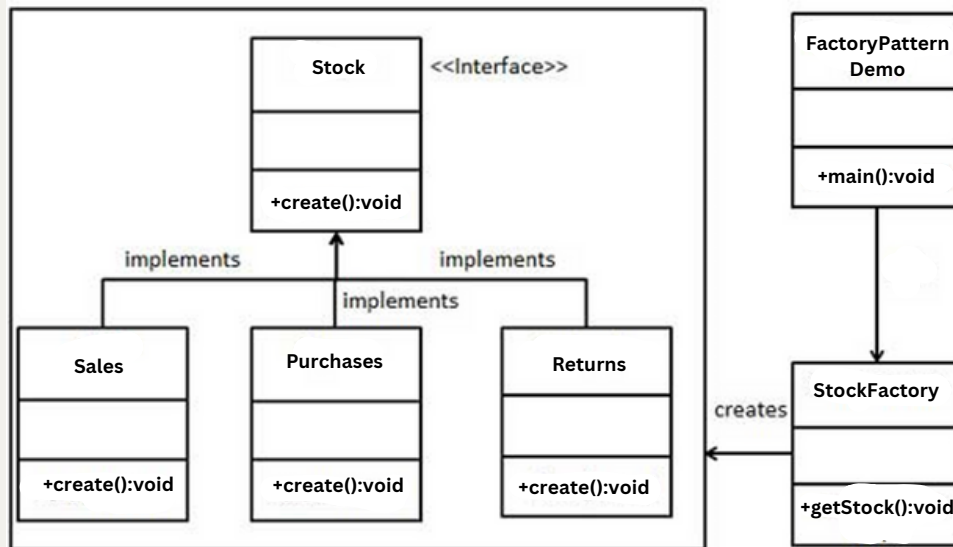


Image: Diagram of Factory design pattern. Tutorialspoint.com. (2023). Available at: [https://www.tutorialspoint.com/design\\_pattern/images/singleton\\_pattern\\_uML\\_diagram.jpg](https://www.tutorialspoint.com/design_pattern/images/singleton_pattern_uML_diagram.jpg) [Accessed 12 Apr. 2023].

## 4.3

### Singleton design pattern

This design pattern limits the number of times a class can be instantiated. Static variables built within it can accommodate distinct, private instances of itself. A class is referred to as a singleton if it only permits the creation of one instance of itself and grants access to it. To obtain a class instance, a singleton class must offer a global access point. Drivers' objects, logging, caching, and thread pools all leverage the singleton pattern ([www.tutorialspoint.com](http://www.tutorialspoint.com). (n.d.)).

#### Structure

Whilst ensuring that only one object is produced, a single class must be in charge of creating an object. The need to instantiate the class is eliminated because the class offers a method of directly accessing the class's sole object (*Refactoring Guru*, 2014).

- **Object creation is managed in this design pattern.** Hence ,static fields will only appear once per class because the total to one is restricted whilst maintaining the freedom to add more as necessary.
- **This design pattern allows data-driven financial decisions** to made. This may interest an accountant as they will be able to record sales and track stock levels. Utilizing the singleton design pattern decreases errors , ensures data correctness and consistency in accounting records. Hence ,this pattern is relevant to mirror data updates when there is only one instance of the class.
- A purchases class with variables may appear more than once in the website , however ,the singleton design pattern assures the head of marketing that the **sales data analysis can be carried out with integrity**. As a result of a class having only one instance , data updates are kept consistent.
- Employing the singleton design pattern guarantees that there is only one instance of the class and allows the store manager to **manage stock levels, make decisions and reorder processes based on reliable information**

([www.tutorialspoint.com](http://www.tutorialspoint.com). (n.d.)).

## How will this design pattern be applied to the project?

- **A class needs to contain a private constructor** to prevent an instance of the class from being created from outside the class. This can then be declared as a singleton. e.g. A "Store" class will define the "getInstance" method.
- **A public static method must be created to return a singleton instance.** e.g. This refers to the "getInstance" method that controls access to Singleton instance.
- **Only one instance of the singleton class must be created** to ensure that the program is thread-safe. e.g. Make sure that no other thread has already started the instance while this one is still awaiting the lock's release. To carry this out, utilize a "ThreadLock" method.
- **The "lazy initialization" must be utilized inside the static method.** On its initial call, the new object should be placed in its static field. On each subsequent call, the procedure must always return that particular instance. e.g. "return store.instance".
- **Utilize the singleton instance by replacing all direct calls** to the singleton's constructor with calls to its static creation function by going through the client code.

([www.tutorialspoint.com](http://www.tutorialspoint.com). (n.d.).

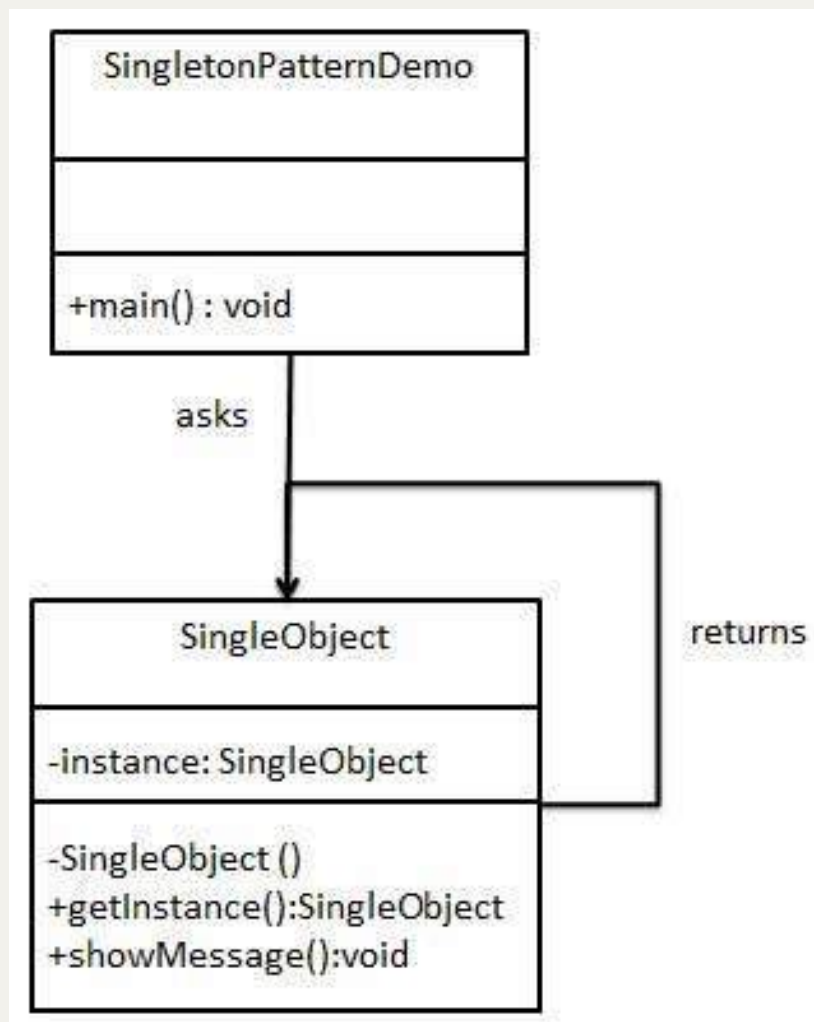
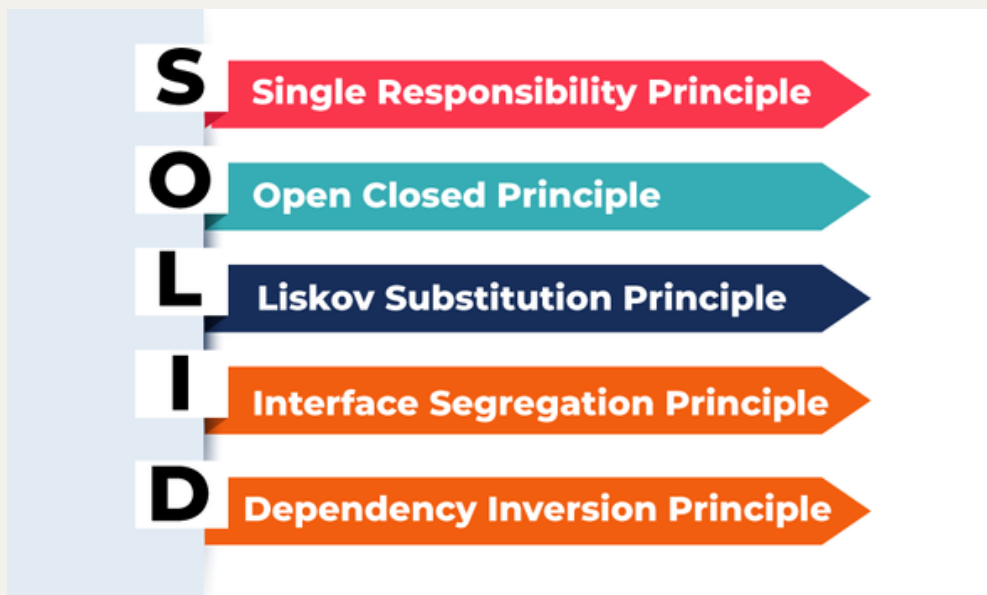


Image: Diagram of Singleton design pattern. Tutorialspoint.com. (2023). Available at: [https://www.tutorialspoint.com/design\\_pattern/images/singleton\\_pattern\\_uml\\_diagram.jpg](https://www.tutorialspoint.com/design_pattern/images/singleton_pattern_uml_diagram.jpg) [Accessed 12 Apr. 2023].

The above design patterns are particularly chosen to ensure that the SOLID principles are taken into consideration.



Geeksforgeeks.org. (2023). Available at: <https://media.geeksforgeeks.org/wp-content/uploads/20220910005416/SingleResponsibility2.png> [Accessed 16 Apr. 2023].

The SOLID principles are a subset of the principles of object-oriented design, which are intended to improve the development process of object-oriented software. The acronym “SOLID” stands for the following five principles:

- Single responsibility principle: Each class should have only one responsibility. According to the SRP, a class should only have one reason to modify. It promotes the idea that a class should only have one duty or goal in order to make it simpler to comprehend, test, and maintain. Classes are kept narrowly focused so that changes to one responsibility won't impact unrelated functions.
- Open-closed-principle: Software units should be expandable without having to change their behavior. Software entities (classes, modules, and functions) are advised to be open for extension but closed for alteration, according to the OCP. It promotes the use of polymorphism and abstraction to enable the addition of new functionality without changing old code. This idea encourages the reuse of code, flexibility, and simpler maintenance.
- Liskov substitution principle: A derived class should always be used instead of its base class. According to the LSP, objects of a superclass should be swappable with objects of its subclasses without impairing the system's integrity. Subclasses must, in other words, abide by the same contract as the superclass. By guaranteeing interoperability between derived classes and their base classes, this approach encourages code expansion and prevents unexpected behaviour.
- Interface segregation principle: Interfaces should be perfectly adapted to the requirements of the accessing clients. The ISP advises against making clients rely on interfaces they do not use. Instead of having a single, expansive interface, it promotes designing specialized interfaces that are tailored to the demands of clients. This principle reduces unneeded dependencies and lessens the impact of changes on customers by keeping interfaces focused and cohesive.
- Dependency inversion principle: Classes on a higher level of abstraction should never depend on classes on a lower level of abstraction. High-level modules shouldn't be dependent on low-level modules, according to the DIP. They ought to both rely on abstractions. By establishing abstractions (interfaces or abstract classes) that specify how modules communicate with one another, this principle promotes decoupling between modules.

**Provide more info on SOLID principles -Expansion on SOLID principles**

(IONOS Digital Guide. (n.d.)).

# Architecture Patterns



## 5.1

### RESTful Architecture

The architectural design that establishes standards across computer systems on the internet is known as the Representational State Transfer architecture. This architecture facilitates system communication as it utilized as as the foundation for communication between the client and server. The stock management website manages incoming and outgoing stock data as well as farmer data and this architecture is able to accomplish non-functional needs such as scalability, performance and extensibility that will satisfy the website. Hence ,it will be simple to build a scalable and adaptable website that uses a defined set of HTTP methods to access and alter data (*Codecademy. (n.d.).*

#### Structure

Communication occurs with the aid of the Hypertext Transfer Protocol on the internet. This system includes a server with the necessary resources as well as a client who makes the resource request (*Codecademy. (n.d.).*

#### Relevance

- Since the RESTful architecture is **platform- and language-independent** ,several devices and platforms that need to access the stock management website will benefit from the use of this architecture. Hence, different scenarios can be tested during the development process because the client and server run independently. This allows the current syntax and platform to adapt. Ultimately , characteristics such as QoS assurances ,security and session management can be added by extending HTTP. This may please the head of marketing.
- Since the client and server are distinct **the website can be scaled without any issues** e.g. an increase in the websites traffic where usage patterns can change significantly during the day or year can be easily accommodated.This can satisfy the head of marketing's expectations.

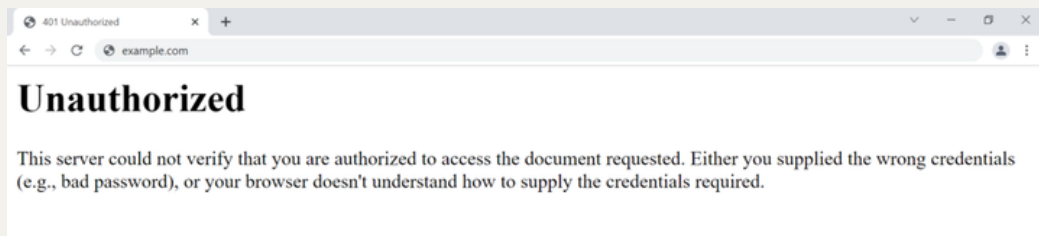
- **Transferring data from one server to another and deploying database changes are made easier** with this architecture. This allows various components of a stock management website to function independently. This may interest the store manager as stock levels are constantly changing.
- The RESTful architecture **makes the combination of other applications with the stock management website is easier** when integrating them.
- Since the RESTful architecture **supports secure communication** between the client and the server , security measures are enhanced when handling sensitive financial and inventory data .This satisfies customers' specific business needs as well as the accountant who requires sensitive data to be secure.

(freeCodeCamp.org,2020).

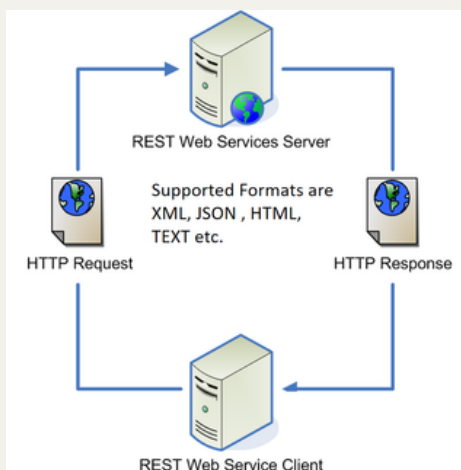
## How will this design pattern be applied to the project?

- Resources such as raw materials, shipments, orders and items are importance components that need to be listed. These **resources will be made available via the websites API**.
- **Each product must be given a URL** that adheres to the RESTful principles. e.g./shipments or /items.
- **HTTP has various methods that will be utilized** in the RESTful architecture. These methods include the GET , POST , DELETE and PUT methods. e.g. The GET method can be used to view a list of items on the /items URL.
- To **safeguard the resources** and stop unauthorized access, utilize authentication and authorization procedures. e.g. Use passwords and limit access to certain roles.
- **Industry standard data formats e.g. JSON or XML must be utilized**. These represent the data transmitted between the client and the server. e.g. Links to the product's related orders and stock levels may be included in a product resource.
- **Standards response codes** e.g. 401 Unauthorized for a request that needs authentication that the web server returns , **must be utilized**.
- To ensure that developers can fully comprehend the set of definitions and protocols being used, **create a document** e.g. This may include a document that describes resources and the parameters that can be supplied to them.

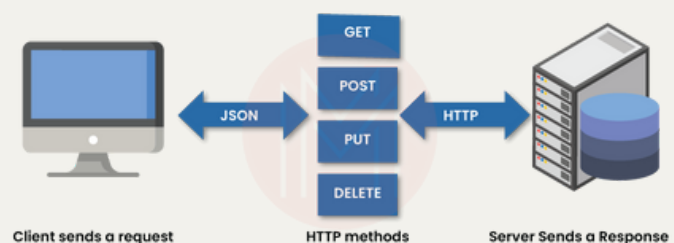
(www.guru99.com. (n.d.).



lonos.com. (2023). Available at: <https://www.lonos.com/digitalguide/fileadmin/DigitalGuide/Teaser/401-Unauthorized-t.jpg>[Accessed April 16 2023].



www.oreilly.com.(n.d.).RESTful web services-Distributed Computing in Java9 [Book]. [online] Available at: <https://www.oreilly.com/library/view/distributed-computing-in/9781787126992/a64ac390-42df-4a08-bb25-c81988170103.xhtml> [Accessed April 16 2023].



IT Technologies, M. (2021). What is REST API | REST API Tutorial. [online] Mindmajix. Available at: <https://mindmajix.com/rest-api-tutorial> [Accessed April 13 2023].



## Model-View-Control architecture

This architecture pattern emphasizes a division between the business logic and appearance of the software as it is a design paradigm for implementing user interfaces, data, and controlling logic. In relation to the stock management website, the model will store the information about incoming and outgoing stock as well as farmer data. The view would allow users such as the head of marketing, store manager and accountant to see information about the stock levels for example. In response to user activities, updating data would be handled by the controller (*Martin, M. (2019)*).

### Structure

User's data-related logic is represented by the Model component and is passed between the View and Controller components. Since controllers server as an interface between the Model and View components, the business logic can be handled by manipulating data using the Model component that interacting with Views to generate the output (*Martin, M. (2019)*).

### Relevance

- Testing the website is made simple as **each component can be tested independently** to ensure the website functions properly. To replicate the various possible system behaviors for unit testing, a portion of each module must be isolated. Hence, even if the websites functionality complexity increases to an extreme, testing and managing the code base is simple.
- The MVC design enables **distinct components to be changed and enhanced independently** of one another without affecting other components, which makes it simple to improve the entire program. Ultimately, the MVC paradigm can facilitate application modification over time. This may be required by the store manager who may require additional features to the website. Hence an improved development process,
- The MVC design allows the website to become more **simple to create and maintain by dividing components independently**. e.g. Storing a significant amount of intricate financial and inventory data is crucial for a stock management website. This **improves the websites scalability and flexibility**. This is vital for the accountant.
- By returning unformatted data **the view engine can be developed and used to create and optimize the user interface** independently of the application's underlying logic. This enhances the users experience and pleases the head of marketing. Hence, the ability to provide multiple views.

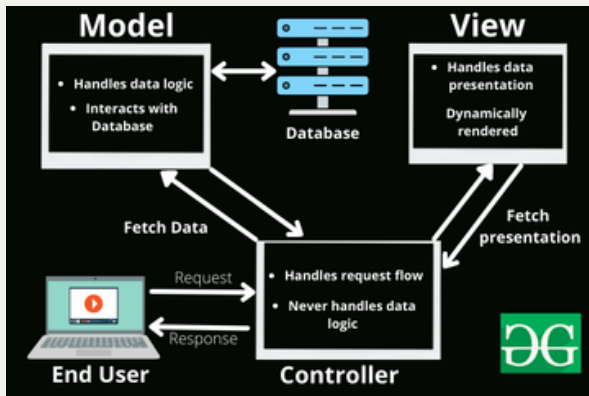
(Anon, (022).

### How will this design pattern be applied to the project?

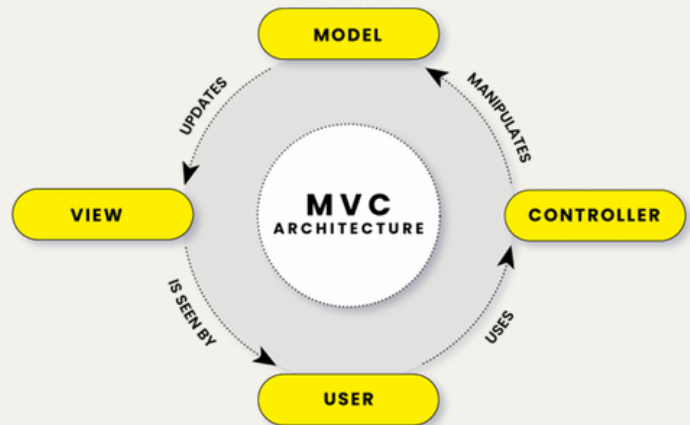
- Stock management website data e.g. prices items and quantities are all maintained by **the model that manages systems business logic**. This involves computing the stock's overall value for the store manager and producing reports like financial documents that are vital for an accountant.
- Users will be able to generate reports, examine and edit stock items, and adjust quantities and prices because the **view shows the data and user interface** on the stock management website. This may aid the store manager in managing stock data.
- After receiving user's input through the View, the model can be updated. This is a result of **the controller managing communication between the model and the view**.
- By utilizing the web browser, **a user can make a request for a list of products**. This request is sent by the browser to the controller. The Controller will then ask the Model to search the database for the list of products. The Model then conducts a database search and provides the Controller with a list of products. If the Controller receives the product list from the Model, the Controller will request that the View displays the product list. After receiving the request, the View renders a list of products and sends it in HTML format to the Controller.

(freeCodeCamp.org, 2021).





GeeksforGeeks. (2022). MVC Framework Introduction. [online] Available at: <https://www.geeksforgeeks.org/mvc-framework-introduction/>. /Accessed April 12, 2023



fireup.pro. (n.d.). Advantages of Using MVC in Application Development - fireup.pro. [online] Available at: <https://fireup.pro/blog/advantages-of-using-mvc-in-application-development/>. /Accessed April 12, 2023

### 5.3

## Event-driven architecture pattern

The flow of an event-driven architecture is built to be controlled by and react to crucial business events e.g. An event could be a transaction that takes place. Handling massive amounts of data and real-time changes may benefit from this method (Amazon Web Services, Inc. (n.d.).

### Structure

The stock management system will react to system-level events through event alerts e.g. the accountant, head of management or shop manager will be informed about an event that updates the inventory and farmer information. This will be triggered when new stock arrived (Sayfan. G. ,2022).

### Relevance

- **Volumes of data and a number of events can be processed** as a result of the distributed event-handling architecture being able to scale. This means that specific components can react to events as they happen without having to wait for other components to complete processing e.g. The store manager may need to process large amount of stock details.
- **Matching consumers' demands for customization is made simple** with this approach. This will benefit the customer experience and management because the website can be easily modified by adding new features without effecting other components. This aids the head of marketing with updating the user's experience.
- **Real-time analytics enables carriers to recognize new patterns in stock levels as they emerge** and actions can be taken as events take play. Since EDA enables components to react to events fast and effectively , the websites performance can be enhanced. This may benefit the store manager and head of marketing by lowering the need for costly processing and storage.
- **The system's fault tolerance is increased** because even if one component malfunctions, others can continue to work without any problems. Additionally, without disrupting the system as a whole, any component can be independently designed, tested, and deployed. By enabling components to handle events independently of one another and minimizing the impact of failures, EDA can increase website reliability. The accountant may require a reliable system when dealing with sensitive financial data.

(Simborg,. M. ,2021 , ww.ibm.com. (n.d.). ).

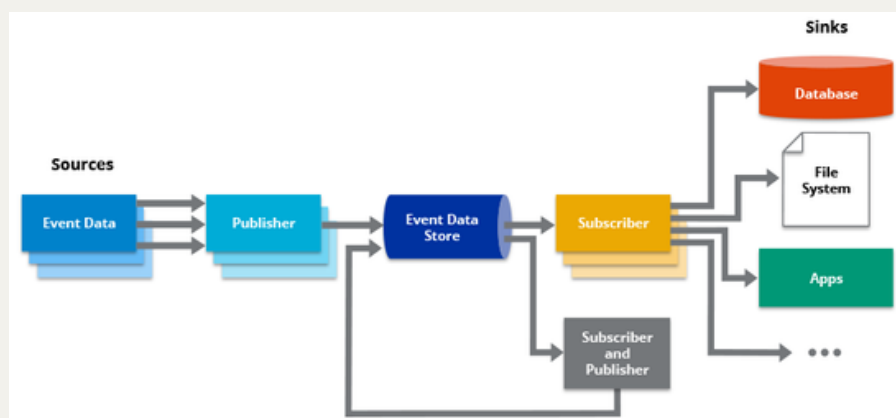
- When a user places , updates or cancels an order , **the order management system creates an event.** Other parts of the system, like the reporting service or the inventory management service, would ingest these events.
- The event-driven architecture could be applied for a service that processes payments for the store manager and creates an event each time one is made.
- The creation of a sales report or change in inventory levels could start an event. The user interface or the reporting program for the accountant could use these events.
- IoT devices can produce events and provide real-time data on various truck characteristics like fuel levels , the GPS or engine specifications. Hence ,when transportation needs to be scheduled, a transportation management system (TMS) that generates an event should be used.
- The user management service or user interface may utilize events like a user account's information changing when they check in or out. This may be useful for the head of marketing to recognize these events.
- The integration of systems such as a supply chain management system , accounting software and point of sale system can benefit the accountant and shop manager .

(wojciech-gebis ,2022)..

### How will this design pattern be applied to the project?

- The system must be able to comprehend and react to events , hence **events such as data , system or user modifications that important to the system must be found** and defined in a precise way.
- **The structure and schema must specified for each event.** These must be adaptable to changes in the future , hence the schema and structure should be extensible and flexible. The data that will be included in the event and the format in which it will be transmitted must both be specified.
- The system's event producers are the elements that create **events that must be published to the relevant channels** .Hence, each event producer needs to be linked with the event-driven architecture.
- **The system components that react to system events** like data storage components or pieces of business logic must be subscribed to the relevant event channels. These are know as **event consumers** and should receive and handle events appropriately.
- The programming that entails **verifying the occurrences, altering the data, and causing the proper reactions to the systems events is known as the event processing logic.** To handle high amounts of events and recovery from errors, the fault-tolerant and scalable event processing logic must be created.
- **Monitoring and optimizing the event-driven architecture** entails observing event channels, spotting performance hiccups, and modifying the system as necessary.This ensures that the implemented event-driven architecture satisfies the system's performance and reliability criteria.

(Awsstatic.com. 2023, blog.hubspot.com. (n.d.).



IdPrice-MSFT(nd).Event-drivenarchitecturestyle-AzureArchitectureCenter.[online]learn.microsoft.com.Availableat:https://learn.microsoft.com/en-us/azure/architecture/guide/architecture-styles/event-driven/AccessedApril12,2023)

# Conclusion

In conclusion , the high-quality solution report underlines the value of satisfying non-functional needs in the creation of the stock management website, in addition to enhancing maintenance and guaranteeing the availability of the newest products. Effectively managing massive volumes of data and user traffic requires performance factors including reaction time and scalability. To safeguard sensitive information and guarantee the system's integrity, security procedures including data encryption and authentication are crucial. The Model-View-Controller (MVC) architecture and the Singleton design pattern, which support code reuse, testability, and maintainability, respectively, are aligned with the SOLID principles. The MVC design divides concerns and makes it easier to maintain and add new features while the Singleton pattern ensures effective use of shared resources. These patterns can help the stock management website provide a dependable, scalable, and secure solution that satisfies the requirements of both the company and the end customers.

*Conclusion is too short - Conclusion has been expanded*



# Bibliography

- White, D. (2019). Transaction Processing System (TPS) | Techfunnel's Complete Guide. [online] Techfunnel. Available at: <https://www.techfunnel.com/fintech/transaction-processing-system/>. (Accessed: April 01, 2023).
- AltexSoft (2019). Non-functional Requirements: Examples, Types, How to Approach. [online] AltexSoft. Available at: <https://www.altexsoft.com/blog/non-functional-requirements/>(Accessed: April 01, 2023).
- Hamilton, T. (2019). Performance Testing Tutorial: What is, Types, Metrics & Example. [online] Guru99.com. Available at: <https://www.guru99.com/performance-testing.html>(Accessed: April 01, 2023)
- Honig, J. 2022. What Makes Software User-Friendly? | DocuWare. Available at: <https://start.docuware.com/blog/document-management/what-makes-software-user-friendly>(Accessed: April 03, 2023)
- Cudd, G. (2023). Benefits of User Research in UX: Why Is It Important? [online] Don't Do It Yourself. Available at: <https://ddiy.co/user-research-in-ux/> [Accessed 15 Apr. 2023]
- McConnell, M. (2018). 9 Principles of Good Web Design - read our guidelines to consider. [online] Feelingpeaky - Creative design agency, London. Available at: <https://www.feelingpeaky.com/9-principles-of-good-web-design/>
- Ed.gov. (2020). Chapter 8-Protecting Your System: User Access Security, from Safeguarding Your Technology, NCES Publication 98-297 (National Center for Education Statistics). [online] Available at: <https://nces.ed.gov/pubs98/safetech/chapter8.asp>(Accessed April ,05 2023)
- Lipan, M. (2021). Website Security Checklist: 18 Ways to Secure Your Site. [online] HOSTAFRICA. Available at: <https://www.hostafrica.co.za/blog/security/website-security/> [Accessed 15 April. 2023]
- Harvey, S. (2020). 8 Secure Coding Best Practices Learned from OWASP. [online] KirkpatrickPrice Home. Available at: <https://kirkpatrickprice.com/blog/secure-coding-best-practices/>.(Accessed:April 09 ,23)
- Anon, (2021). Why Is Accounting Reliability So Important for Your Business? - Franco Blueprint. [online] Available at: <https://francoblueprint.com/accounting-reliability/>(Accessed:April; 07,23.)
- www.javatpoint.com. (2011). Software Engineering Software Reliability Metrics - javatpoint. [online] Available at: <https://www.javatpoint.com/software-engineering-software-reliability-metrics>(Accessed April 08, 23).
- Creately Blog. (2011). Tech Talks ~ The Importance of Reliability. [online] Available at: <https://creately.com/blog/development/tech-talks-the-importance-of-reliability-2/>(Accessed : April 12,2023)
- ReQtest (2020). What is Reliability Testing & Why It Is Important in Software Testing? [online] ReQtest. Available at: <https://reqtest.com/testing-blog/reliability-testing/#:~:text=Reliability%20testing%20helps%20to%20ensure> [Accessed 12 Apr. 2023].
- Solutions, M. (2022). How to Ensure Accurate Data Entry. [online] Magellan Solutions. Available at: <https://www.magellan-solutions.com/blog/how-to-ensure-accurate-data-entry/> [Accessed 13Apr.il 2023].
- Shen, S. (2019). 7 Steps to Ensure and Sustain Data Quality. [online] Medium. Available at: <https://towardsdatascience.com/7-steps-to-ensure-and-sustain-data-quality-3c0040591366/>(Accessed:Aoril 13, 2023)
- www.linkedin.com. (n.d.). What are the best practices for training users on new software features and updates? [online] Available at: <https://www.linkedin.com/advice/1/what-best-practices-training-users-new-software#:~:text=Training%20users%20on%20new%20software%20features%20and%20updates%20is%20a> [Accessed 16 Apr. 2023]
- Insights, M. (2018). Importance of Training and Support After Software is Implemented (In a COVID-19 Corporate Environment). [online] Mindfield Consulting Corp. Available at: <https://mindfieldconsulting.com/importance-of-training-and-support-after-software-is-implemented/#:~:text=When%20adopting%20or%20creating%20new> [Accessed 14 Apr. 2023].
- www.digitalocean.com. (n.d.). Observer Design Pattern in Java | DigitalOcean. [online] Available at: <https://www.digitalocean.com/community/tutorials/observer-design-pattern-in-java>.(Accessed:April 11, 2023)
- www.tutorialspoint.com. (n.d.). Design Patterns - Observer Pattern - Tutorialspoint. [online] Available at: [https://www.tutorialspoint.com/design\\_pattern/observer\\_pattern.htm](https://www.tutorialspoint.com/design_pattern/observer_pattern.htm)(Accessed:April 12, 2023)
- Caballero, C. (2021). Understanding the Observer Design Pattern. [online] Medium. Available at: <https://betterprogramming.pub/understanding-the-observer-design-pattern-f621b1d0b6c9> [Accessed 16 Apr. 2023]
- www.tutorialspoint.com. (n.d.). Design Patterns - Observer Pattern - Tutorialspoint. [online] Available at: [https://www.tutorialspoint.com/design\\_pattern/observer\\_pattern.htm](https://www.tutorialspoint.com/design_pattern/observer_pattern.htm)(Accessed:April 12, 2023)
- www.tutorialspoint.com. (n.d.). Design Pattern - Factory Pattern - Tutorialspoint. [online] Available at: [https://www.tutorialspoint.com/design\\_pattern/factory\\_pattern.htm](https://www.tutorialspoint.com/design_pattern/factory_pattern.htm)[Accessed 1Apr il 13 2023]
- Refactoring Guru (2014). Factory Method. [online] Refactoring.guru. Available at: <https://refactoring.guru/design-patterns/factory-method>
- Stack Overflow. (n.d.). Factory Pattern. When to use factory methods? [online] Available at: <https://stackoverflow.com/questions/69849/factory-pattern-when-to-use-factory-methods> [Accessed 1Apr il 13 2023]
- IONOS Digital Guide. (n.d.). Factory pattern: the key information on the factory method pattern. [online] Available at: <https://www.ionos.com/digitalguide/websites/web-development/what-is-a-factory-method-pattern/#:~:text=The%20factory%20pattern%20aims%20to> [Accessed April 13 2023]
- Csulb.edu. (2023). Available at: <https://home.csulb.edu/~pnguyen/cecs277/lecnotes/factory> [Accessed April 14. 2023].
- www.tutorialspoint.com. (n.d.). Design Pattern - Singleton Pattern. [online] Available at: [https://www.tutorialspoint.com/design\\_pattern/singleton\\_pattern.htm#:~:text=This%20pattern%20involves%20a%20single](https://www.tutorialspoint.com/design_pattern/singleton_pattern.htm#:~:text=This%20pattern%20involves%20a%20single) [Accessed April 15. 2023]
- Refactoring Guru (2014). Singleton. [online] Refactoring. Guru. Available at: <https://refactoring.guru/design-patterns/singleton>[Accessed April 14 2023]
- www.tutorialspoint.com. (n.d.). Design Pattern - Singleton Pattern. [online] Available at: [https://www.tutorialspoint.com/design\\_pattern/singleton\\_pattern.htm#:~:text=This%20pattern%20involves%20a%20single](https://www.tutorialspoint.com/design_pattern/singleton_pattern.htm#:~:text=This%20pattern%20involves%20a%20single) [Accessed April 15. 2023]
- IONOS Digital Guide. (n.d.). Factory pattern: the key information on the factory method pattern. [online] Available at: <https://www.ionos.com/digitalguide/websites/web-development/what-is-a-factory-method-pattern/#:~:text=The%20factory%20pattern%20aims%20to> [Accessed 16 Apr. 2023].
- Codecademy. (n.d.). What is REST? [online] Available at: <https://www.codecademy.com/article/what-is-rest>.(Accessed:April 13 , 2023)
- freeCodeCamp.org. (2020). The Benefits of Going RESTful - What is REST and Why You Should Learn About It. [online] Available at: <https://www.freecodecamp.org/news/benefits-of-rest/>

# Bibliography

- [www.guru99.com](https://www.guru99.com/restful-web-services.html). (n.d.). RESTful Web Services Tutorial with Example. [online] Available at: <https://www.guru99.com/restful-web-services.html>[Accessed April 10, 2023].
- Martin, M. (2019). MVC Tutorial for Beginners: What is, Architecture & Example. [online] Guru99.com. Available at: <https://www.guru99.com/mvc-tutorial.html>(Accessed April 14, 2023)
- Anon, (2022). 6 Benefits Of Using MVC Model For Effective Web Development. [online] Available at: <https://profoundedutech.com/blog/6-benefits-of-using-mvc-model-for-effective-web-application-development/>.(Accessed April 13 2023)
- freeCodeCamp.org. (2021). How the Model View Controller Architecture Works - MVC Explained. [online] Available at: <https://www.freecodecamp.org/news/model-view-architecture/#:~:text=How%20MVC%20Architecture%20works>.(Accessed: April 15 2023)
- Amazon Web Services, Inc. (n.d.). Event-Driven Architecture. [online] Available at: <https://aws.amazon.com/event-driven-architecture/#:~:text=What%20is%20an%20Event%2DDriven> [Accessed 16 Apr. 2023]
- Sayfan, G. (2022). Introduction to event-based programming. [online] Aiven.io. Available at: <https://aiven.io/blog/introduction-to-event-based-programming>.(Accessed April 15, 2023)
- Simborg, M. (2021). The Four Advantages of an Event-Driven Architecture. [online] Volt Active Data. Available at: <https://www.voltactivedata.com/blog/2021/03/event-driven-architecture/#:~:text=3.-> [Accessed April 10., 2023]
- [www.ibm.com](https://www.ibm.com/topics/event-driven-architecture). (n.d.). Event-Driven Architecture | IBM. [online] Available at: <https://www.ibm.com/topics/event-driven-architecture>Accessed April 10. 2023.
- wojciech-gebis (2022). An In-Depth Guide to Event-Driven Architecture: What It Is, How It Works, and Why You Need It. [online] nexocode. Available at: <https://nexocode.com/blog/posts/guide-to-event-driven-architecture/> [Accessed 16 Apr. 2023]
- Awsstatic.com. (2023). Available at: [https://d1.awsstatic.com/product-marketing/EventBridge/1-SEO-Diagram\\_Event-Driven-Architecture\\_Diagram.b3fbc18f8cd65e3af3ccb4845dce735b0b9e2c54.png](https://d1.awsstatic.com/product-marketing/EventBridge/1-SEO-Diagram_Event-Driven-Architecture_Diagram.b3fbc18f8cd65e3af3ccb4845dce735b0b9e2c54.png) [Accessed 16 Apr. 2023]
- [blog.hubspot.com](https://blog.hubspot.com/website/event-driven-architecture). (n.d.). What Is Event-Driven Architecture? Everything You Need to Know. [online] Available at: <https://blog.hubspot.com/website/event-driven-architecture/>Accessed: April , 12 2023





# Thank You

[www.itsolutions.com](http://www.itsolutions.com)