Requirements “El gran abarrotero”

Brayan Campos

Github: braycampos (Brayan Campos) (github.com)

1.- Global understanding of the problem (project context): The Mexican grocery store chain "El gran abarrotero" has grown significantly in recent years. This chain started with just one store, and as it was a family business, purchases from distributors and producers were manually recorded in a notebook, as were the profits. Now that it has become a chain of grocery stores, keeping such records in a notebook is no longer the most optimal approach.

2.- Candidate Requirements:

Branches
Employees
Merchandise
Customers
Sales
Supliers

3.- Funtional Requirements Capture:

- **Branch Management:** The system must allow the creation, updating, consultation, and deletion of branch records.
  It must be possible to associate each branch with its respective employees, merchandise, and sales.

- **Employee Management:** The system must allow the creation, updating, consultation, and deletion of employee records.
  Each employee must be associated with a specific branch.
  The system must allow the assignment of roles to employees, such as cashier, manager, etc.

- **Merchandise Management:** The system must allow the creation, updating, consultation, and deletion of merchandise records.
  It must be possible to associate each merchandise item with a supplier and a specific branch.
  The system must automatically update merchandise stock after each sale.

- **Customer Management:** The system must allow the creation, updating, consultation, and deletion of customer records. It should be possible to associate each customer with the sales they have made.

- **Sales Management**: The system must allow the creation of sales records, associating the sale with a customer, the sold merchandise, and the corresponding branch.
  It must allow querying sales by date, branch, customer, and merchandise.
  The system must automatically calculate the total sale amount.

- **Supplier Management:** The system must allow the creation, updating, consultation, and deletion of supplier records. Each supplier must be associated with one or more merchandise items.

4.- Non – Functional Requirements Capture:

- **Security:** The system must ensure data security by implementing role-based access controls. It must include encryption for sensitive information, such as customer and supplier contact details.

- **Performance**: The system must be able to handle at least 1,000 simultaneous sales transactions without degrading performance. Data queries should return results in less than 3 seconds for a data volume of up to 100,000 records.

- **Usability:** The system interface must be intuitive and easy to use, allowing users to perform their tasks with minimal training. It should be accessible from different devices, including tablets and smartphones.

- **Scalability:** The system must be scalable to support store growth, allowing the addition of new branches, employees, and products without requiring significant system modifications. It must be possible to integrate the system with other business applications, such as accounting or inventory management systems.

- **Availability**: The system must be available at least 99.9% of the time, allowing users to access it at any time.

5.- Associated with Functional Requirements:

- **Branch Management – Security**: Only administrators should be able to create, update, or delete branch records.
- **Employee Management – Usability**: The process of adding or updating employee information should be simplified, using predefined forms and field autocompletion.
- **Merchandise Management – Performance**: The system must update merchandise stock in real-time without affecting sales processing speed.
- **Sales Management – Availability**: The system must be available throughout the store's opening hours, allowing sales to be recorded at any time.

- **Supplier Management – Scalability**: It must be possible to add new suppliers and associate multiple products with each supplier without affecting system performance.

6.- Capture of Non-Functional Requirements Not Associated:

- **Compatibility**: The system must be compatible with multiple operating systems, including Windows, macOS, and Linux. It must function correctly in major web browsers (Chrome, Firefox, Edge, Safari).
- **Portability**: The system must be portable, allowing deployment in different environments (on-premises or in the cloud) without significant modifications. It must allow data migration between SQL databases without data loss or functionality loss.
- **Maintainability**: The system code must follow well-defined coding standards to facilitate maintenance and future updates. There must be detailed system documentation, including guides for installation, configuration, and troubleshooting.
- **Reliability**: The system must include automatic backup mechanisms to ensure that data is not lost in case of system failures. It should have a mean time to recovery (MTTR) of less than 2 hours in case of a critical failure.
- **Scalability**: The system must be able to handle a 100% increase in the number of users and transactions without requiring significant reengineering. It must be easily integrable with new technologies or modules, such as an advanced inventory management system or an e-commerce platform.
- **Interoperability**: The system must be able to exchange data with external systems, such as payment platforms or accounting systems, using standard APIs. It must support integration with third-party services using standard protocols such as REST or SOAP.
- **Sustainability**: The system must be designed to minimize resource consumption, such as memory and processing, optimizing hardware usage. It must include design practices that allow easy hardware upgrades and responsible disposal, minimizing environmental impact.
- **Accessibility**: The system must comply with Web Content Accessibility Guidelines (WCAG 2.1), ensuring it can be used by people with disabilities. It must include options to increase contrast, support for screen readers, and keyboard navigation.
- **Privacy**: The system must comply with data protection regulations (e.g., GDPR) to ensure the privacy of users' personal information. It must include mechanisms for anonymization and pseudonymization to protect sensitive data.
- **Availability**: The system must guarantee 99.9% uptime, including high availability and redundancy strategies. It must include real-time monitoring systems to detect and mitigate issues before they affect users.