# Informatics Engineering

Leandro Faria, Luis Silva, Muhammad Umair, Pedro Melo

January 2023

## 1 Contextualization and Motivation

The need for project management is obvious. Projects are complex, and require careful planning and execution to be successful. Unfortunately, large companies that provide project management programs often require clients to purchase additional modules, such as invoicing. This can add unwanted complexity and cost to the project management process. It would be great if there was a complete project manager that did not need additional modules to function properly.

Project management is becoming increasingly important as organizations look for ways to manage costs, improve efficiency and productivity, and complete projects on time and within budget. With the right project management system, organizations can improve project visibility and ensure that tasks are completed on time while staying within budget. A complete project manager would make project management easier, faster and more effective.

Project managements are an essential part of any successful organization. They provide direction, guidance and support throughout the process of planning, monitoring and controlling a project. With the right tools and processes, project managements can help ensure the successful completion of projects. Having a complete project management that does not require extra modules would make it easier for organizations to achieve their goals.

Organizations that have adopted project management systems have reported increased success rates in due projects.

## 2 Requirements Analysis

The concept of requirements in engineering is a crucial one and is essential for the successful development of any system or product. It represents a clear description of the functionality that users need of any constraints that must be taken into account when designing and implementing a system.

## 2.1 Functional Requirements

The functional requirements describe the system's capabilities and should be met at a later stage in the application development, as early as possible. These requirements play an important role in software engineering, where their success depends on their clarity and objectivity. They also depend on the type of application being built, as well as user expectations, along with the domain knowledge of the project team. They should not include technical details unrelated to any particular technology used during development; henceforth, it allows for exploration of diverse technologies throughout the project.

| Cultural and Organizational | | |
|---|---|---|
| ID | Description | Date |
| FR01 | The system must allow user login | 0 |
| FR01 | The system must allow the administrator to consult the user list | 0 |
| FR02 | The system must allow the administrator to create a new user | 0 |
| FR03 | The system must allow the administrator to edit a user | 0 |
| FR04 | The system must allow the administrator to consult the details of a user | 0 |
| FR05 | The system should allow the administrator to remove a user | 0 |
| FR06 | The system must allow you to consult the warehouse list | 0 |
| FR07 | The system must allow for the creation of a new warehouse. | 0 |
| FR08 | The system must allow editing a warehouse | 0 |
| FR09 | The system must allow consulting details of a warehouse | 0 |
| FR010 | The system must allow removing a warehouse | 0 |
| FR011 | The system must allow querying the supplier list. | 0 |
| FR012 | The system must allow the creation of a new supplier. | 0 |
| FR013 | The system must allow editing a supplier. | 0 |
| FR014 | The system must allow consulting details of a supplier. | 0 |
| FR015 | The system must allow you to remove a supplier. | 0 |
| FR016 | The system must allow consultation of a list of products. | 0 |
| FR017 | The system must allow the manufacture of new products | 0 |
| FR018 | The system must allow the modification of an product | 0 |
| FR019 | The system must allow consulting details of an product. | 0 |
| FR020 | The system should allow removing an product. | 0 |
| FR021 | The system must allow the entry of new products in a specific warehouse. | 0 |
| FR022 | The system must allow the exit of products from a specific warehouse. | 0 |
| FR023 | The system must allow the exchange of products between warehouses. | 0 |
| FR024 | The system must allow adjustments in the stock of products. | 0 |
| FR025 | The system must allow the verification of products. | 0 |
| FR026 | The system must allow you to create a new project. | 0 |
| FR027 | The system must allow you to edit a project. | 0 |
| FR028 | The system must allow you to consult projects. | 0 |
| FR029 | The system must allow you to consult details of a project. | 0 |

## 2.2 Non-Functional Requirements

Non-Functional Requirements are of great importance for ensuring that a system meets the needs and expectations of its users. At these encompass all the needs independent of system functionalities, they establish the properties and constraints of a system. It should be noted that any failure in meeting these requirements can lead to the system becoming useless. According to Robertson and Robertson (2006), these non-functional requirements can be classified into eight types - appearance, usability, performance, operability, maintainability and support, security, cultural and organizational, legal. For instance, in order to ensure proper performance and responsiveness, maximum response times need to be established. Similarly, other aspects such as usability and maintainability also need to be taken into account while designing and developing a system.

### 2.2.1 Appearance

The appearance requirements are of paramount importance when it comes to satisfying the demands imposed by customers. It is essential to take into account the use of certain colors, shapes and sizes in order to meet the visual expectations of customers. Moreover, having too stringent appearance requirements can prove to be a detriment to success as they may not fit the product that is being created. It is important to strike the right balance between meeting the customer's needs, while also taking into consideration the practicality of the product itself. In any case, appearance requirements play a key role in customer satisfaction and should no be overlooked.

| Operational | | |
|---|---|---|
| ID | Description | Date |
| NFR01 | The product must have a graphical interface with cool colors. | 0 |
| NFR02 | The product must follow a responsive layout to allow integration with various types of devices. | 0 |
| NFR03 | The system should allow users to register on the platform through a simplified interface. | 0 |

### 2.2.2 Usability

Usability requirements are essential to ensure a successful user experience. By understanding how the system will be used, we can create more effective and efficient usability criteria. This information is gathered from the customer and helps to determine how easy the system will be to use. To ensure that our system meets all usability requirements, we must evaluate the effectiveness of our designs and make sure they are up-to-date with the latest trends and user expectations. Usability testing is an important step in this process to ensure the usability of our system. Being aware of current industry standards and trends also helps us create a system that meets both the user's needs as well as their

expectations. By considering these facts, we can create a product with excellent usability that customers will appreciate.

| Operational | | |
|---|---|---|
| ID | Description | Date |
| NFR04 | The product must take into account easy accessibility | 0 |

### 2.2.3 Performance

The performance requirements are essential in order to achieve the desired functional requirements. These performance requirements can include speed, system availability, storage capacity, and other factors. It is important that these performance requirements be enforced in order to maintain the highest levels of operation for the system. The performance requirements also have a direct impact on the system's ability to meet customer needs and demands. In order to ensure the best results, these performance requirements should be monitored regularly and any shortcomings addressed swiftly in order to ensure the highest level of functionality.

| Operational | | |
|---|---|---|
| ID | Description | Date |
| NFR05 | The product must have search engine optimization (SEO). | 0 |
| NFR06 | The product must be accessible 24 hours a day, 7 days a week. | 0 |
| NFR07 | The product must have an SQLServer database as persistent storage. | 0 |

### 2.2.4 Operational

The operational requirements set out by this system will provide the necessary structure and framework for its successful functioning. This description of the system will include all the components which are essential in ensuring its smooth operating. Furthermore, it is also important to consider other factors that may influence the performance of the system, such as external environment and its interaction with third-party systems. Without a comprehensive understanding of these elements, it is impossible to guarantee that the system will function as intended. Therefore, it is vital to ensure that the operational requirements are clearly established in order to maximize its efficiency and reduce any potential risks.

| Operational | | |
|---|---|---|
| ID | Description | Date |
| NFR08 | The product must be made available on a web platform. | 0 |
| NFR09 | The system must be supported by most browsers. | 0 |

### 2.2.5 Maintainability and Support

Maintainability and Support are vital components that help ensure a system's longevity. These requirements guarantee the system will remain adaptable over time and be able to easily accommodate improvements and repairs. They give the system a certain level of agility, allowing it to remain competitive in the market even as new technologies arise. Without maintainability and support requirements, a system is likely to quickly become outdated and unable to keep up with current industry standards.

| Cultural and Organizational | | |
|---|---|---|
| ID | Description | Date |
| NFR10 | The product must work locally in case of Cloud access failure. | 0 |

### 2.2.6 Security

Meeting the security requirements is a key element for any system in order to prevent malicious access attempts. By currently implementing these requirements, the system can guarantee the confidentiality and integrity of its data, as well as other attributes. Security is a priority that should not be overlooked and should be taken into account when designing systems. Taking the right security measures can protect the system from potential threats and ensure that no unauthorized parties can gain access.

| Cultural and Organizational | | |
|---|---|---|
| ID | Description | Date |
| NFR11 | The product must provide a hierarchy of access to the platform. | 0 |
| NFR12 | The product must allow for the caching of data in case of loss of internet connection. | 0 |
| NFR13 | The product shall perform personal data encryption on the user's personal device. | 0 |
| NFR14 | The product shall perform personal data encryption on the user's personal device. | 0 |
| NFR15 | The system must provide a token system for automatic user authentication. | 0 |
| NFR16 | The personal data of each user can only be accessed by the user himself or by the administrator. | 0 |
| NFR17 | Users' personal information must be encrypted. | 0 |

### 2.2.7 Cultural and Organizational

The cultural and organizational requirements are essential for any successful system integration. By describing the stakeholders, it is possible to identify their habits and preferences and integrate a system in such a way that it fits

into the existing culture and political landscape. This requires a keen understanding and sensitivity to the existing atmosphere, which can be achieved by engaging with stakeholders and understanding their needs. This helps create an environment which is conducive to a successful integration of systems into the cultural and organizational context.

| Cultural and Organizational | | |
|---|---|---|
| ID | Description | Date |
| NFR18 | The product must be available in the English Language. | 0 |

### 2.2.8 Legal

It is crucial to ensure that the system is properly put into operation, and it must comply with all legal requirements. Therefore, it is essential to focus on the laws, regulations and norms that the system needs to follow. This will help ensure that the system operates in accordance with current laws and regulations, and the legal requirements are met. Furthermore, it is important to have a sound understanding of the legal implications of any changes or amendments to the system, in order to ensure that these changes are compliant with the existing laws.

| Legal | | |
|---|---|---|
| ID | Description | Date |
| NFR19 | The product must comply with the guidelines of the General Data Protection Regulation. | 0 |
| NFR20 | The system must use cookies in order to provide a better user experience. | 0 |

# 3 User profile

A user can either be an administrator or a regular user.

The administrator user is the highest level of user and as such has privileges that the regular user does not. With the administrator user, they are able to view the list of employees using the program and manage their accounts. This provides an efficient way to manage employee information and provide updates accordingly. Furthermore, the administrator user can check the working hours of each employee and review the financial status of the company through the dashboard.

The regular User is able to perform most actions within the program including viewing active projects and their budgets, customer list, stock, and dashboard. As a regular User, they are able to monitor project progress and review the customer list for any necessary updates or changes. Additionally, they can view stock to see what needs ordering or restocking and check the

dashboard for any important information regarding finances or other reports related to the company.

Both users provide essential roles in keeping track of key information within the program. By having both an administrator and regular user, it ensures that all necessary tasks are completed efficiently with accurate reporting.

# 4 Use Case

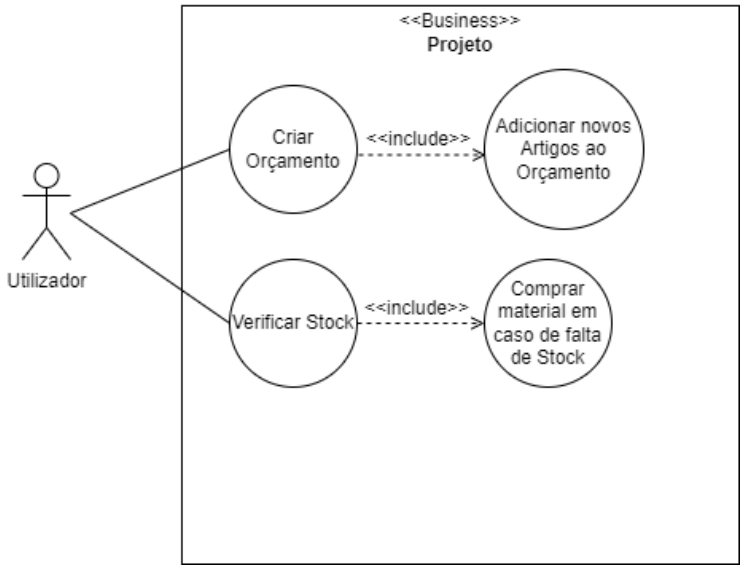


Figure 1: Create a Project

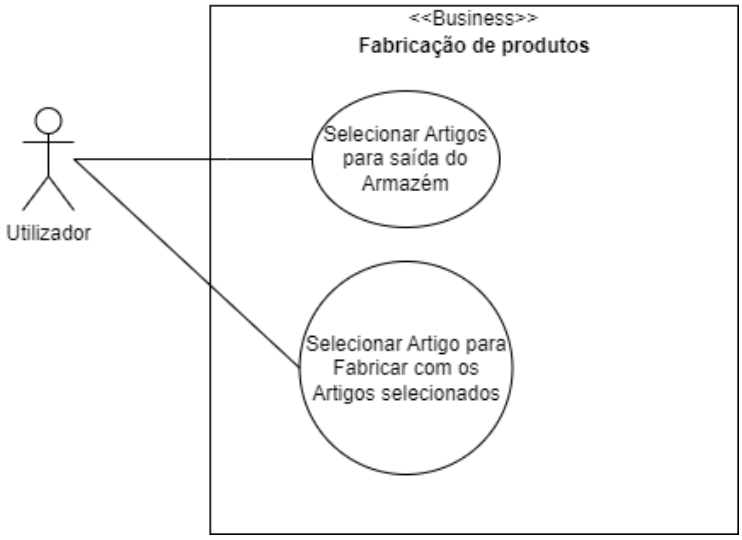


Figure 2: Create a Product

# 5 Entity Relationship Diagram

The Entity Relationship Diagram (ER) shown in the image below provides us with a simplified overview of the database. It establishes the relationship between various entities within our system, which are used to store, manage and retrieve data. This ER allows us to have a better understanding of how the various components of our database are connected and how they work together. The ER also provides an easier way to identify and resolve any potential issues related to the database. In addition, alongside with this report there is a file attached to that provides a more detailed and comprehensive version of the Entity Relationship Diagram, which is not possible to fit in this report due to its size. This more extensive version can be used to gain an even greater understanding of the database and how it is structured.

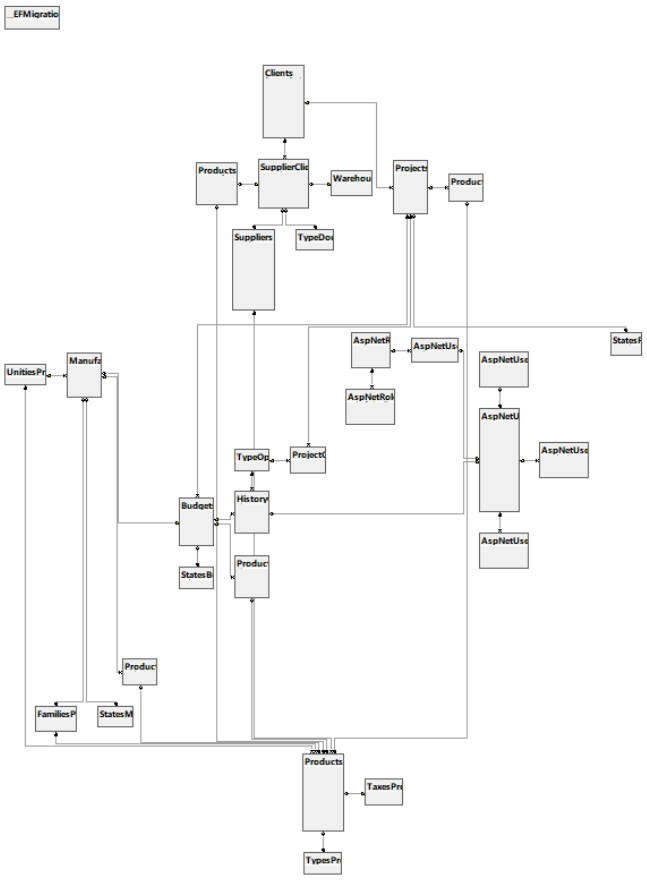


Figure 3: Entity Relationship Diagram

# 6 Architecture & Implementation

The Model-View-Controller (MVC) architecture is a software design pattern that helps developers create robust and maintainable applications. it is composed of three primary layers: The Model layer, The View layer and the Controller Layer. The Model Layer is responsible for validating and manipulating the data, as well as reading and writing it. This layer contains business logic, which ensures the integrity of the data. The View Layer is responsible for representing this data to the user in an appropriate form. This layer should contain no business logic, but instead represent a representation of how the user should interact with their data. Finally, the Controller Layer controls all information flow that goes through the system. It defines what information to generate, which rules to apply and where the information should go. The Controller

Layer acts as a bridge between the Model Layer and the View Layer, allowing for seamless communication between them.

MVC is a power architecture that offers numerous advantages over traditional user interface development strategies. By creating a clean separation between different aspect of an application, we can ensure logical organization and allow for greater scalability and flexibility in our code base. Additionally, by organizing the application into different layers we can also improve readability and re-usability of our code-base while also improving maintainability.

By applying the Model-View-Controller (MVC) pattern to the solution, the controllers and models are placed in the API, while the data presentation layer is placed in the web client. This allows for increased flexibility and scalability of the application, as well as ensuring that any changes made to either layer do not affect each other. Additionally, there may be utility classes that are contained in the project solution if required. The MVC pattern provides an effective way of organizing the components of an application, making it easier to debug or modify in the future.

## 6.1 Data Base

The SQLServer was used for our database and the entity-relationship model was employed to describe the data and business process information in an abstract way. It is important to note that the model consists of two main components, namely the entities and heir relationships. It is a systematic way of describing and defining a business process that can be implemented in a database. The data is represented using symbols such as rectangles for entities and diamonds for relationships. This model helps us to understand the communication between different entities and how they interact with each other. It also aids in data modeling which includes validation of the data and verifying its accuracy, thus resulting in higher efficiency of data processing. Overall, it is clear that SQLServer provides many benefits to organizations looking to store their data securely and efficiently.

# 7 Conclusion

The development of this solution allowed the group to acquire numerous useful skills for the future, both academic and professional. Starting with learning how to build RESTful API, this being a programming interface with a set of programming patterns that allow the construction of applications and their use. We also learned how to work with MVC architecture patterns. In some parts of the project Javascript language was used, something that was also taken as a great learning opportunity. All these skills proved to be important stepping stones into a greater understanding of the underlying technology and its implementation.

The conclusion is that such projects are essential for enhancing our knowledge and understanding of the technology and its application. The development process of this project enabled us to gain knowledge in many areas ranging from RESTful APIs to MVC architectures and JavaScript languages. This has proven to be extremely beneficial not only academically but also professionally as it has provided us with the necessary skills to further our career prospects in this field.