DOI: https://doi.org/10.33571/rpolitec.v21n42a1

DESIGN AND CONSTRUCTION OF A LEGO PALLETIZER PROTOTYPE

Ingrid Durley Torres-Pardo[1], Moises Naranjo-Jaraba[2], Valentina Sánchez-Osorio[3], Julián Ramírez-Espinosa[4], Nicolas Arango-Vergara[5], Nicole Tatiana Rondón-Avendaño[6]

[1] Ph.D. in Engineering, Faculty of Engineering Lecturer, SYSLAC research group leader. Universidad Católica Luis Amigó, Medellín-Colombia. Email: ingrid.torrespa@amigo.edu.co, ttps://orcid.org/0000-0003-4503-7512
[2] Industrial Engineering student, member of the SYSLAC research group, Universidad Católica Luis Amigó, Medellín-Colombia. Email: moises.naranjoja@amigo.edu.co, https://orcid.org/0009-0007-3264-4627
[3] Industrial Engineer, member of the SYSLAC research group, Universidad Católica Luis Amigó, Medellín-Colombia. Email: valentina.sanchezso@amigo.edu.co, https://orcid.org/0009-0006-6535-9278
[4] Systems Engineering student, member of the SYSLAC research group, Universidad Católica Luis Amigó, Medellín-Colombia. Email: julian.ramirezpi@amigo.edu.co, https://orcid.org/0009-0006-8144-8369
[5] Mechatronics assistant, Systems Engineering student, member of the SYSLAC research group, Universidad Católica Luis Amigó, Medellín-Colombia. Email: nicolas.arangove@amigo.edu.co, https://orcid.org/0009-0004-1648-4991
[6] Industrial Engineering student, member of the SYSLAC research group, Universidad Católica Luis Amigó, Medellín-Colombia. Email: nicole.rondonav@amigo.edu.co, https://orcid.org/0009-0000-3776-0116

RESUMEN

This article describes the design and construction process of an industrial palletizer prototype developed to solve the logistic requirements of a furniture manufacturing company. The process of research, design, assembly and programming of the prototype is detailed, as well as the main challenges encountered and the solutions implemented. For the materialization of the design, the LEGO MINDSTORMS EV3 kit was used together with Python programming, achieving a semi-automated system capable of simulating the industrial palletization process. The prototype adapts to objects of different dimensions, regulating the speed and height of the operation. In addition, the mechanical challenges of assembly with LEGO parts and the optimization of iterations to improve functionality are described. The results show the feasibility of the prototype as a basis for larger-scale industrial systems, with potential applications in various sectors. It is concluded that the developed solution is functional, practical and scalable for process automation in industry.

Keywords: Prototype; mechatronic palletizer; robotics; LEGO

Recibido: 19 de mayo de 2025. Aceptado: 11 de noviembre de 2025
Received: May 19, 2025. Accepted: November 11, 2025

DISEÑO Y CONSTRUCCIÓN DE PROTOTIPO LEGO DE UNA PALETIZADORA

ABSTRACT

Este artículo describe el proceso de diseño y construcción de un prototipo de paletizadora industrial, desarrollado como solución a los requerimientos logísticos de una empresa de fabricación de muebles. Se detalla el proceso de investigación, diseño, ensamblaje y programación del prototipo, así como los principales desafíos encontrados y las soluciones implementadas. Para la materialización del diseño se utilizó el kit LEGO MINDSTORMS EV3 junto con programación en Python, logrando un sistema semiautomatizado capaz de simular el proceso de paletización industrial. El prototipo permite adaptarse a objetos de distintas dimensiones, regulando la velocidad y la altura de la operación. Además, se *describen los retos mecánicos del ensamblaje con piezas LEGO y la optimización de iteraciones para mejorar la funcionalidad. Los resultados del estudio evidencian la viabilidad del prototipo como base para sistemas industriales de mayor escala, con aplicaciones potenciales en diversos sectores. Se concluye que la solución desarrollada es funcional, práctica y escalable para la automatización de procesos en la industria.*

Palabras clave: *Prototipo; paletizadora mecatrónica; robótica;*

Cómo citar este artículo: I. Torres, M. Naranjo, V. Sánchez, J. Ramírez, N. Arango, N. Rondón. "Design and construction of a LEGO palletizer prototype", Revista Politécnica, vol.21, no.42 pp.09-24, 2025. DOI:10.33571/rpolitec.v21n42a1

1. INTRODUCCIÓN

Nowadays, automation [1] has become a fundamental pillar both in everyday life and in the business world, ranging from large corporations to small and medium-sized enterprises. Its implementation enables the optimization of processes [2], reduction of time, and increase in productivity. Technological advancements have made it possible to incorporate automation into all aspects of industry, transforming the way daily activities are carried out and significantly increasing efficiency.

The ability of modern technologies to integrate systems and processes has revolutionized production and task management, ushering in an era in which automation is not only possible but necessary to remain competitive. However, many companies still rely solely on manual methods for essential tasks such as product packaging, limiting their efficiency and competitiveness compared to other companies that do implement automation in their operations.

This article presents the design and construction of a palletizer prototype developed using the LEGO MINDSTORMS EV3 [3] set, known for its support in facilitating innovative educational experiences for the training of professionals in the design and management of manufacturing systems, addressing the lack of access to real production systems [1], and the Python programming language, which enabled the development of the system software, achieving functionality, accessibility, and adaptability to various needs.

The prototype [4] allows for the packaging of products with different dimensions and characteristics, thereby demonstrating its ability to adapt to a dynamic and changing industrial environment. Furthermore, it offers an effective way to address a specific task that is often repetitive and physically demanding for workers.

The initiative emerged from the problem and need identified in a wooden furniture manufacturing company, where the packaging process was performed manually. This process not only required a large amount of time but also caused ergonomic issues for the operators. Repetition, poor posture, and physical strain caused by manual processes result in inefficiency and pose significant health risks for workers.

Automating this task was proposed as an improvement opportunity in the logistics area and as a viable solution for the packaging process, with the aim of optimizing workflow, increasing productivity, and reducing physical risks associated with poor posture from load handling. The result was an effective and flexible solution, through a palletizer prototype capable of carrying out the packaging of various objects, allowing the automation of this task. This initiative not only seeks to solve a specific problem but also aspires to serve as a replicable model for other industrial sectors.
Once tested and validated for its purpose, this prototype can serve as a guide for building a full-scale palletizer, depending on the needs of each company. Thanks to its versatility in packaging, the prototype can be applied to different industrial sectors, offering an adaptive solution that can be adjusted to various conditions and requirements.

Throughout the article, the different stages involved in the development of the palletizer prototype are described, from the conceptualization of the need, the methodology used, and the phases of the process involved, to the construction of the prototype and its software. The challenges faced in each stage and the solutions implemented to overcome them during the prototype's development are also presented. Finally, the results obtained are presented, including how the functionality of the prototype was evaluated and tested, and its potential industrial applications and future improvements are explored, highlighting its potential as a reference for the design of a full-scale version.

2. THEORETICAL FRAMEWORK

This section presents the main concepts related to the industrial palletizer prototype described herein.

Palletizing
Palletizing is a key logistical operation within the supply chain, where products are grouped on pallets (flat platforms made of various materials such as wood, plastic, or metal) to facilitate handling, storage, and transportation. According to López and Martínez (2018) [5], palletizing improves warehouse space efficiency, reduces loading and unloading times, and optimizes operating costs. This process is crucial to ensuring the safe handling and transport of large volumes of products in industrial and commercial sectors.

Packaging
Packaging is the process through which products are prepared for transportation and storage, using materials that protect them during handling. Packaging not only ensures the integrity of the products but also enhances their presentation to the end customer. According to González and Pérez (2020) [6], packaging is essential in all phases of the supply chain, as it protects products from impacts and environmental factors, thereby improving transport efficiency.

Pallet
A pallet is a structure, typically made of wood, plastic, or metal, that allows products to be grouped and handled more efficiently during transportation and storage. This device facilitates the use of loading machinery such as forklifts and optimizes space during transport. According to Castro and Hernández (2019) [7], in the palletizing process, products are arranged in a way that maximizes available space and minimizes the risk of damage during transportation.

Software Programming
This involves designing algorithms and implementing sequential or conditional commands to control devices or systems. According to García and Torres (2020) [8], in the industrial field, the software used in automated systems enables process optimization, such as palletizing control, by adjusting system behavior based on the needs of the production process. In this project, programming in Python facilitates the automation of the product stacking process.

Prototype
The development of prototypes with LEGO Technic has also enabled the construction of SCARA-type robotic arms, providing students with a didactic tool to understand their industrial applications. This prototype, based on kinematic models and programmed with LEGO Mindstorms NXT 2.0, simulates industrial processes in an accessible manner (Escobar Viveros, Gomez Jaramillo & Ruiz Castillo, 2019) [9]. A prototype is a preliminary version of a product or system created to assess its functionality before final implementation. According to Fernández (2019) [10], prototyping is essential in engineering project development, as it allows for the identification of failures and early design improvements. In this case, the palletizer prototype was designed to simulate and automate the stacking process in a controlled environment, validating its effectiveness prior to final implementation.

Prototyping with LEGO MINDSTORMS EV3
LEGO MINDSTORMS EV3 (LEGO, 2026) [11] has commonly been used for product prototyping; due to the variety of pieces, their articulation, and the presence of a programmable brick, it enables the creation of functional models of products, robots, and automated systems in general. The use of mechanical, adjustable, interlocking, and easily programmable electrical components is a standout feature of LEGO

products. According to Sánchez and Ramírez (2021) [12], LEGO MINDSTORMS is an ideal tool in both educational and industrial settings, as it facilitates the creation of flexible prototypes adaptable to different needs, allowing for the testing of robotics and automation concepts in an accessible environment.

In the work developed by Toapanta Tipanta (2014) [13], the design and construction of a prototype for what they call a “modular and accessible wrapping machine” for small businesses is proposed, as market alternatives are often expensive due to their complexity. The design includes a rotating platform, a detachable film-holder tower for easier transport, and a pre-stretching system that optimizes plastic usage. Tools such as AutoCAD Mechanical, Autodesk Inventor, and MDSolids were used to validate calculations and select suitable materials, and a cost analysis was conducted to ensure market viability and competitiveness. Although the related work also incorporates a financial perspective by performing a cost analysis, it shares a common goal with this article: to create an accessible prototype for packaging products with plastic film; it should be noted, however, that the design and other characteristics are significantly different.

3. PROTOTYPE DESIGN

The initiative arose from the need to automate the packaging process in a company dedicated to the manufacturing of wooden furniture. However, the prototype has been successfully adapted for use in various industrial sectors, thanks to a versatile design that allows packaging of objects with different dimensions.

Initially, the architecture of the palletizer was inspired by commercial models available online. This approach facilitated an understanding of the general operation of existing palletizers, which allowed for the identification of the essential components of the machine. A key element is the rotating base, which enables the piece to spin on its own axis while being wrapped with protective material. This analysis helped to identify the most important functions of the packaging process.

The development of the prototype was structured into three main phases, as shown in Figure 1. Development Methodology:

Contextualization:
The first stage of development involved defining concepts and reviewing existing work related to palletizing and product packaging. This was done to guide the proposed solution to the problem identified in the kitchen furniture industry, as previously mentioned.

Mechanical Design:
A rotating base driven by a motor was created to allow smooth rotation of the object. In addition, a support tower was designed to transport the packaging material along the surface of the object, achieving a precise and uniform wrap.

Programming and Control:
A control system based on Python was implemented to regulate the speed and height of the operation. This system ensures the adaptability of the prototype, allowing for efficient handling of a wide range of packaging scenarios.

Testing and Adjustments:
Once the initial prototype was developed, a series of rigorous tests were carried out to evaluate its performance under different conditions. These tests focused on analyzing the stability of the rotating base, the precision of the wrapping mechanism, and the responsiveness of the control system. Objects of various sizes and weights were packaged to verify the prototype's adaptability and consistency. Based on the results obtained, adjustments were made to optimize motor speed, packaging material tension, and control parameters, ensuring reliable operation and high-quality packaging in diverse industrial scenarios.

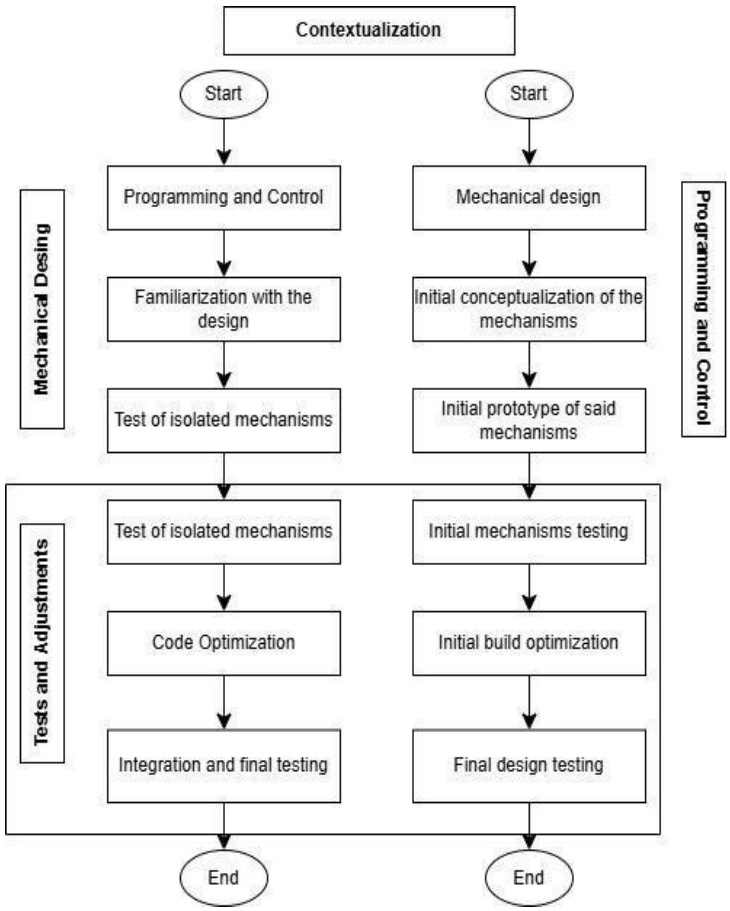

Figure 1. Development Methodology
Source: Authors

The figure above, shows the process in which the project developed, having some contextualization of the issue to give the solution to, comprehend the industry and understand where we could help, then, we started in two development fronts, the system front, which had the programming and control of the motors and sensors involved in the prototype, the familiarization of the design, testing isolated mechanisms to fine tune said components, then, some code optimizations to said mechanisms to make them function as best as possible, and then, integrate it with the physical prototype, which itself, had its process of development.

Said development started, with an initial mechanical design, having an initial conceptualization of the mechanisms, to understand their function, then, the team proceeded to develop some initial prototypes of said mechanisms, they were tested in parallel of the system testings, then, the physical prototypes had some optimizations made, to make them function better, and at the end, to have some final testings integrating the whole system to have it function as a whole integral system.

4. MECHANICAL DESIGN

For the design of the palletizer, shown in Figure 2, initial research was conducted on the packaging process and palletizing systems to understand their operation. Based on this research and images of some industrial products, a turntable-type palletizer was designed, incorporating adjustment mechanisms and programming capabilities. Initially, the end user interacts through software developed and implemented on a programmable brick, which controls the device's operation. This software is stored on an SD card programmed from a computer; the brick maintains a direct connection with the palletizer to execute the instructions issued by the software.

To develop the prototype addressing the identified problem, resources from the LEGO Mindstorms Education EV3 Set 45544 were used, as it is an educational tool that combines building blocks with programming in areas such as engineering, technology, and computer science. It includes a wide variety of pieces, motors, sensors, and a programmable brick, which acts as a processor enabling the control of operations over the components. This makes it possible to build more robust structures, providing the freedom to design as needed.

Based on the information gathered, the palletizer prototype was designed considering two main components: the base, where the assembly elements are located, and the tower, which includes a mechanism that transports stretch film (vinipel) to wrap the products. The first component, the base, uses a motor to rotate the platform on its own axis, so that, as it turns, the object is wrapped around itself. Simultaneously, the tower holds a roll of stretch film that moves up and down, adjusting to the height of the object. As the product rotates, it is wrapped progressively until it is completely covered with stretch film.

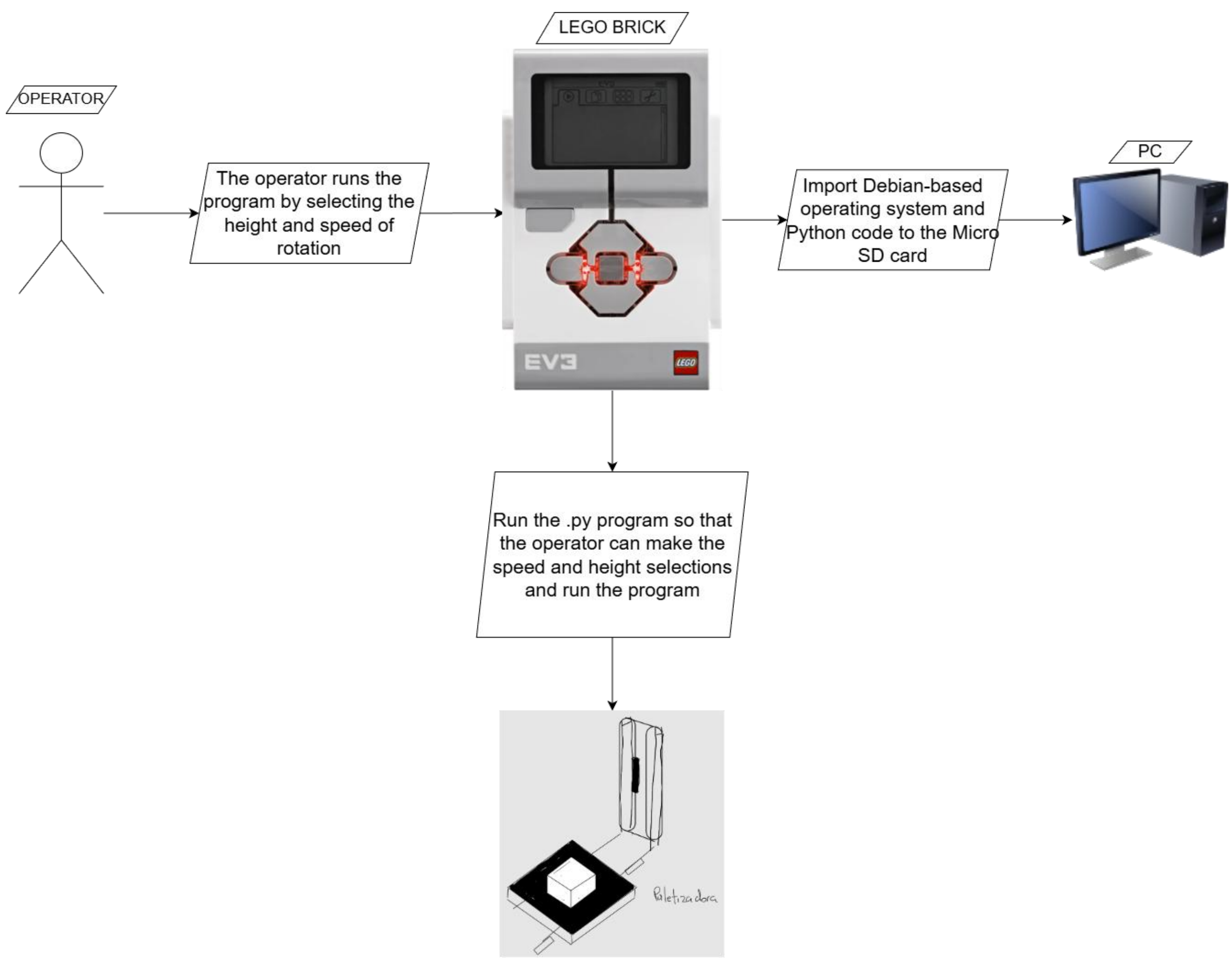

Figure 2. Pälletizer Macro-Architecture Diagram
Source: Authors

The base design includes a motor that allows it to rotate on its own axis. Initially, a preliminary version was developed, as shown in Figure 3, in which the motor transmitted movement through a shaft connected to a worm gear. To transfer the motion to the platform, another shaft with a gear was added, which made contact with the vertically positioned worm gear. This configuration allowed the motion to be transmitted from the motor to the platform.

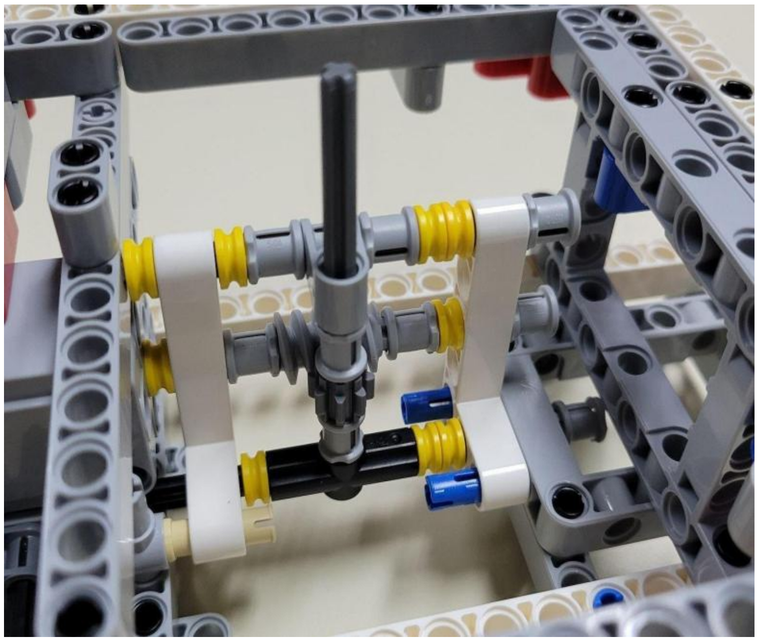

Figure 3. Rotation Mechanism
Source: Authors

However, the design presented several issues, such as a loss of force, due to the fact that the gear and worm gear components shown in Figure 3 did not fit together properly. With use, these pieces tended to shift, which negatively affected the system's performance. In an attempt to prevent this displacement and ensure proper support for the components, additional structural pieces were added. However, this solution was not considered efficient in the long term, as the mechanism and components did not function properly as support for the motor and the upper platform.

As a result, it was necessary to modify the system, starting by replacing the motor with a large motor capable of transmitting rotation directly to the platform. This eliminated the need for a mechanism with multiple parts and the associated loss of power. Figure 4 shows the new design, in which the motor is connected directly to a shaft, allowing the motion to be transmitted efficiently and directly, without losses due to friction or part displacement. Additionally, Figure 5 shows a base that supports both the upper platform and the motor, without requiring a large number of components.

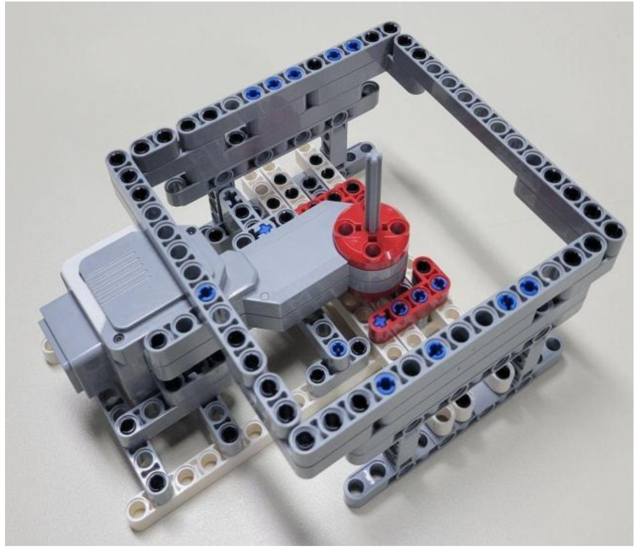

Figure 4. Base
Source: Authors

For the platform, several designs were developed with the aim of allowing the palletizing process to adapt to elements of different shapes and sizes, such as boxes, cylinders, and even flat surfaces simulating the doors manufactured by the company that inspired this project. Ultimately, a flat platform was created, as shown in Figure 5, with a design consisting of several flat pieces positioned horizontally and secured at the corners. These pieces rest on the base, also shown in Figure 5, which provides support for the weight of the items and meets the requirement of adapting to the needs of the process.

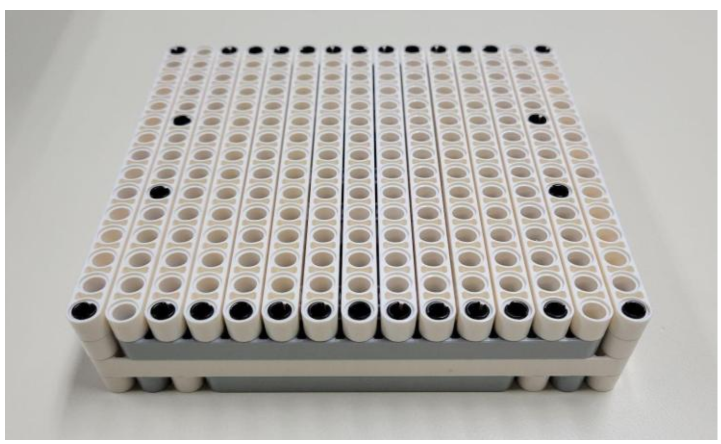

Figure 5. Platform
Source: Authors

At the bottom of the platform, a centrally positioned shaft is connected to the main motor to enable the platform's rotation. As previously mentioned, to ensure the platform's stability, a support base was designed, which can be seen in Figure 5.

For the tower, shown in Figure 6, a design was created to allow the stretch film to be mounted on a belt that moves up and down, covering the entire surface of the objects to be packaged. To enable the movement of the belt, the medium motor was placed at the bottom of the tower, connected to a rail on which the rotating base is mounted. Additionally, two side walls were constructed for the tower, extending upward from the base, and were joined at the back with white beams that connect both sides. To mount the gears, transverse shafts with toothed wheels were installed at the top, middle, and bottom, allowing a circuit of components to rotate around them.

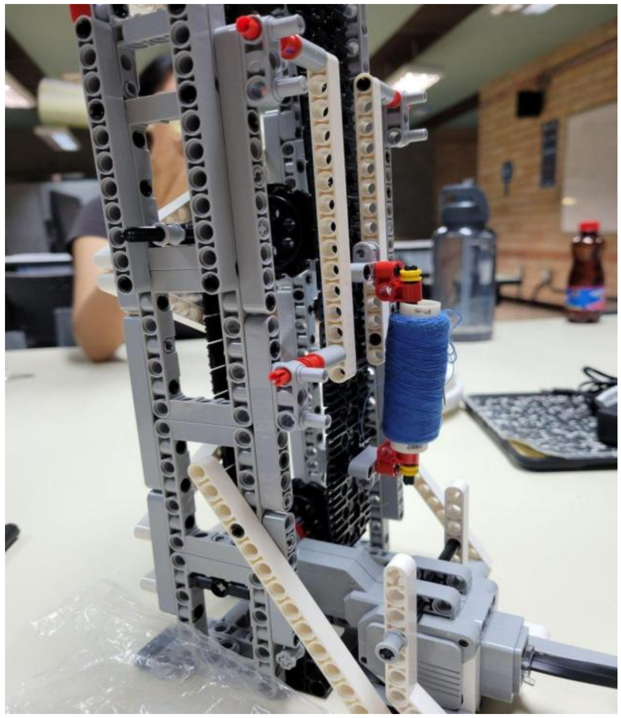

Figure 6. Tower
Source: Authors

In the circuit, a shaft was installed to hold the roll of stretch film; however, during the wrapping process, the film tended to move away from the tower, causing the circuit to detach from the toothed wheels especially after multiple uses. To solve this problem, two lateral beams were added to secure the shaft with the stretch film roll during packaging, as shown in Figure 7, in front of the tower's displacement circuit. Finally, the base was positioned on the rail connected to the tower, ensuring that the rotating base remained in place so the tower could move without interfering with the base's rotation, as shown in the initial design in Figure 7.

Figure 7. Initial Desing
Source: Authors

Once the design of the first prototype shown in Figure 7 was completed, several tests were conducted to verify its functionality and that of each of its components. Among these tests, various pieces of different sizes were used, allowing the palletizing of shapes such as cylinders and boxes. This helped to evaluate the movement of the tower and the displacement of the circuit when palletizing objects of varying sizes, and to determine whether there were any difficulties in its path. Additionally, the performance of the base was evaluated under different weights, as well as the platform's ability to adapt to various shapes. These tests, shown in Figure 8, were conducted using objects such as thermoses and boxes.

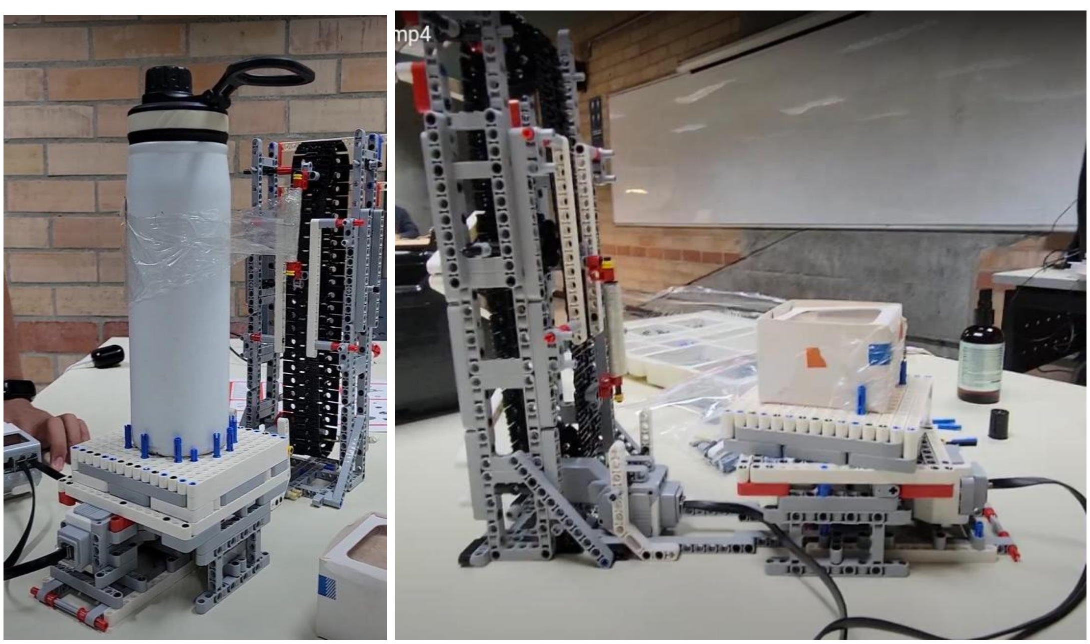

Figure 8. Images of the palletizer in testing

Source: Authors

The results of these tests revealed several issues: the rotating base was too heavy due to the number of components, as shown in Figure 9. Additionally, the extra weight from the objects caused the platform to rub against the base, creating friction and damaging the parts. On the other hand, in the tower, the central gear did not function properly because, during movement, the circuit would shift away from the central gear as a result of the force exerted while palletizing an object. Furthermore, after each test, it was necessary to manually reposition the stretch film at the top of the tower to its initial position. Therefore, it was considered necessary to define a starting position for the stretch film after each cycle in order to optimize the palletizing time.

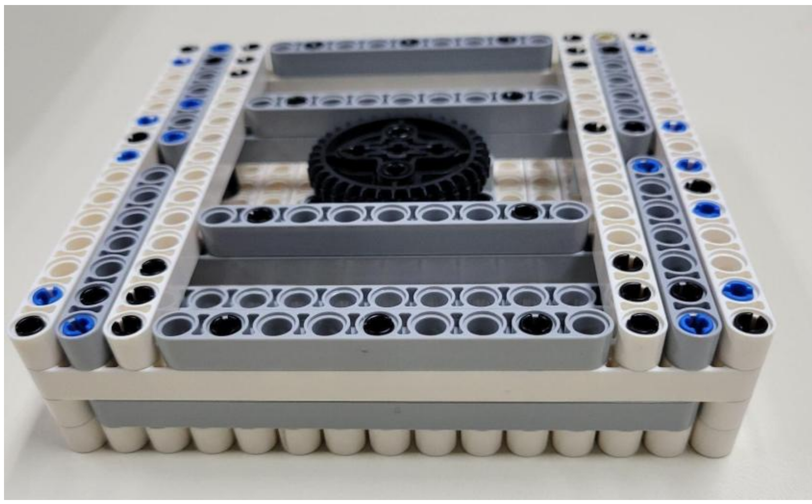

Figure 9. Correction of Parts
Source: Authors

As a result, efforts were made to reduce the number of components and correct the errors in the first design. In the tower, the central gear was replaced with a smaller one and positioned outside the circuit to generate tension, improving the upward and downward movement of the stretch film and optimizing the tower's design by reducing the number of parts. The stability of the toothed wheels was also reconsidered, and their mounting was redesigned to prevent unnecessary movement.

To solve the issue of the stretch film’s initial position, a pressure sensor was added at the bottom of the circuit, next to the motor, as shown in Figure 10. This addressed the previously mentioned issue by enabling the circuit to move to the sensor at the start of the program, thereby establishing the initial position of the stretch film and defining its movement limits. The space within the circuit for palletizing was also adjusted so that the pressure sensor provides an ideal range to fully wrap the items.

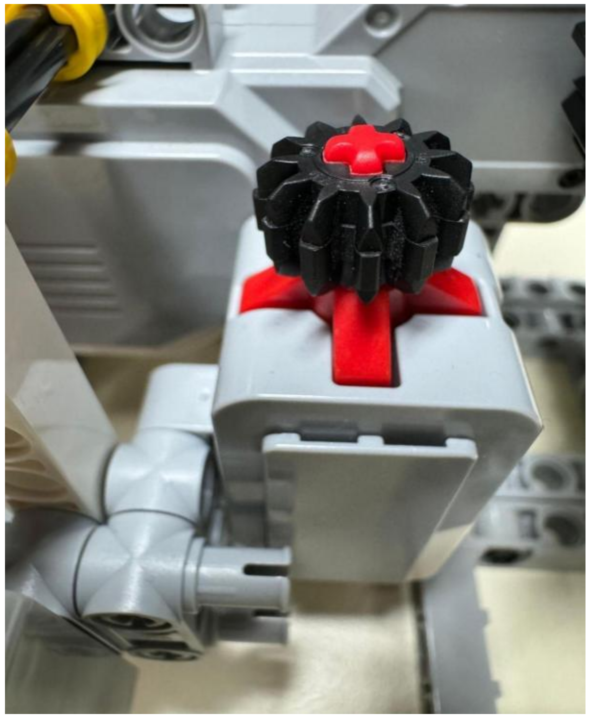

Figure 10. Pressure Sensor
Source: Authors

Regarding the base, to address the issues of weight and friction, a reduction in the number of components was evaluated, and the repositioning of parts was carried out to minimize rubbing without compromising stability. Additionally, the design was corrected to ensure that the base was properly aligned with the motor, eliminating unwanted movements during rotation. The final version of the palletizer prototype is shown in Figure 11.

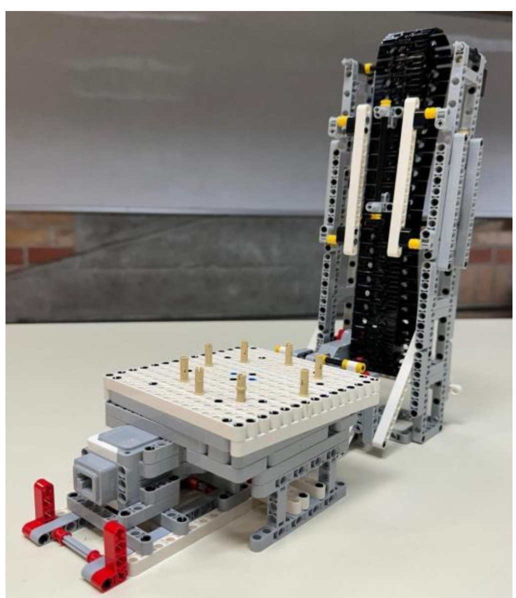

Figure 11. Final Design
Source: Authors

5. PROGRAMMING STRATEGY

To program this prototype, the first step was to understand the operation of a palletizer that packages finished products, analyzing how it works, the behavior of the different parts of the mechanism, and how this could be applied to the main function of the machine. The system was designed to include various modes of operation, such as selectable speed and the height of the object to be palletized the latter being adjusted through a parameter in the code. This parameter was calculated using pre-measured rotation data obtained by observing motor behavior, since each motor has its own specific rotation system (i.e., each motor has a rotation ratio depending on its type). Another aspect considered was the rotation of the base motor for proper packaging (palletizing) of the object. Speed and time measurements were taken to provide flexibility in the available options and avoid being limited to a single function. This versatility allows the machine to package various products. The software was developed primarily through prototyping theory and techniques, taking into account the different components and their potential limitations during construction. The development of the software was based on the use of the Python programming language, which is compatible with the Debian-based Linux operating system used for ensuring compatibility with the LEGO MINDSTORMS EV3 kit.

5.1 Software

This software was developed primarily through prototyping theory and techniques, taking into account the various components and their potential limitations during construction. To maintain coherence in the software structure, it is based mainly on Python code, which can be executed on the LEGO brick through an operating system built on a Linux distribution specifically Debian (GitHub, n.d.) [14]. This operating system decodes and processes the Python code of the prototype and subsequently executes it.

5.2 Programming Strategy

To program this prototype, the first step was to study the operation of a real machine that palletizes goods—understanding how it works, the behavior of its different mechanical components, how the system functions, and how this could be applied to the main functionality of the prototype. The machine was designed to have different modes of operation, including parameters such as selectable speed (low speed and high speed) and the height of the object to be palletized (low, medium, and high height). This flexibility allows for multiple configuration options, avoiding limitation to a single setting.

For the coding process, different approaches were explored. Initially, it began as a simple test script for individual functionalities, where the base rotation was tested separately from the movement of the belt that drives the stretch film. Later, a unified code was developed to integrate both functionalities into a single software. After that, the first features for speed selection were programmed, followed by the coding of height options. Throughout this entire process, several failures and iterative changes occurred. However, the main structure of the code consists of the following key points:

- **Modular Structure:**

The code is divided into specific functionalities that encapsulate well-defined tasks, making it easier to maintain and reuse. This means that any function can be called when needed without having to recreate it from scratch. These well-defined, encapsulated functions such as moving the stretch film, stopping the film, selecting the base speed, among others, make the code reusable and easy to maintain. The coordination of the main cycle is handled through a while True conditional loop, which continuously verifies that the required condition is met during the software's operation.

- **Main Loop:**

An infinite loop structure acts as the main controller, executing tasks sequentially while managing the overall flow of the program.

- **Error Handling:**

The software uses a block designed to capture exceptions, errors, and failures that may occur during program execution. These may include issues such as a motor malfunction, failure to detect motors or sensors, or obstructions in the stretch film. The system ensures that all prototype components are stopped in case something unexpected occurs. When an error is detected, the software immediately halts the process, thereby protecting the hardware. In an industrial setting, this feature would also serve as a safety mechanism for the operator handling the machine or, in the case of a fully automated system, as protection against serious mechanical damage.

5.3 Techniques Used

- **Event-Based Programming:**

User inputs are detected through the physical buttons of the EV3 controller (the brick that hosts both the program and the operating system that manages everything). Therefore, the brick itself is currently the only way to interact with the system. However, in the future, remote control could be implemented to operate the system without relying on the brick, or a touchscreen interface could be added. System interruption management is handled by completely stopping the program, allowing errors to be corrected and the process to be restarted from the beginning. This prevents subsequent operations from being affected by issues caused by the initial error, and it allows for the customization of parameters within the software to manage preconfigured settings.

- **Hardware Control:**

By directly integrating the motors and the pressure sensor with the libraries used in the code, specifically the ev3dev library along with various imported modules for motors, sensors, and buttons functions are synchronized so they begin simultaneously. This includes calibration of the stretch film position using a dedicated touch sensor located at the bottom.

- **Use of Functions to Avoid Redundancy:**

Each action is encapsulated in its own function, so that when a task needs to be executed, it is not rewritten but simply called through the corresponding function. This reduces abstraction and increases code clarity.

- **Time Management:**

The sleep function is used in the code to introduce pauses, enabling synchronization of hardware operations.

- **Controlled Iteration:**

With the use of for loops and while conditions, precise repetition is ensured, preventing the program from running longer than necessary and ensuring it only performs specific actions until a predefined condition is met.

The code handles key actions for its proper functioning, such as the selection of speed and height, providing the operator or user of the prototype with options so that the user experience can be better adjusted to their needs, and in itself, the main functionality of the code also includes an initial verification that the stretch film is at the indicated height before starting the process, avoiding palletizing failures or a poorly executed process, it executes a cycle of repeated movements, preventing unexpected situations such as an extra movement, or as previously mentioned, a poorly performed process, lastly, hardware protection is implemented, with error exceptions, the code, upon detecting any failure in the program's execution flow, immediately stops everything, avoiding problems with the hardware.

5.4 A Technical Formulation

Here, options and techniques were studied to develop the palletizing process, taking into account the different ways to palletize a product and/or pallet (in some future idea or in the scaling of the prototype to a larger concept). Also, during the assembly process, in which the code also began to be developed jointly, challenges were discovered, such as finding a balance in how fast the base could rotate to avoid instability during the operation of the prototype, or even with the test elements, preventing failures such as falling due to centrifugal force, which is the force that drives a body away from its axis of rotation. Furthermore, since this aspect was not initially considered when the rotation was first programmed, the code had to be adapted to prevent such failures.

It was also considered that the stretch film, which moves linearly along the axis, had to operate at a constant and not too high speed, as this could cause problems during palletizing tests with tall objects. Therefore, the speed had to be adjusted so that the actions performed by the prototype would not create issues.

5.5 Software

The software used in this development is a combination of the Linux operating system for the reprogramming of the Lego brick, so that it can understand the programming language chosen for this development, which was Python, using the ev3dev2 libraries (GitHub, 2024) [15], which were used for the communication between the software (in this case Python) and the hardware used (Lego Mindstorms EV3 NXT Kit), so that through structured programming the programmed actions are executed.

6. RESULTS

The tests carried out demonstrated that the prototype is capable of performing the palletizing process efficiently. Some limitations were observed regarding the stability of the rotating base and the weight distribution of the objects, which required adjustments to the mechanical design. Additionally, the implementation of a pressure sensor improved the system's accuracy, optimizing its functionality.

The development of the prototype made it possible to validate the feasibility of implementing an automated palletizing system using accessible resources. It is concluded that this solution can serve as a foundation for future industrial-scale developments, incorporating improvements in stability, speed, and automated object detection.

7. FUTURE WORKS

In future work, the end-user experience could be enhanced through the implementation of part detection algorithms based on computer vision. This would allow for the complete automation of the height and speed selection process, eliminating the need for a graphical interface or any direct interaction with the LEGO brick. In this way, the system would be able to autonomously configure the different operating modes, further simplifying and streamlining the process.

Additionally, integration with more advanced control strategies could be implemented, for example, using a Linear Quadratic Regulator (LQR), or the possible application of Kalman filters to improve both the precision and stability of the rotating base's movement, as well as the accurate detection of objects through a camera that could be integrated into the prototype for automatic object detection, as mentioned by Hughes, Willetts, and Kryczka (2020) [16] in LQG controller for the LEGO MINDSTORMS EV3 Gyroboy Segway robot.

For a future industrial implementation, a user manual could be developed containing the specifications for both the construction and software development, allowing for the creation of a full-scale machine tailored to the specific needs of each company. This approach, centered on artificial intelligence, ensures a more efficient and fully automated operation.

8. ACKNOWLEDGEMENTS

We would like to express our gratitude to the institution Universidad Católica Luis Amigó, Medellín campus, especially to the Department of Basic Sciences for their support in the development of this work by allowing us to use their facilities and tools.

9. BIBLIOGRAPHIC REFERENCES

[1] Sid-Lakhdar, Mokhtar Nizar, Benderbal, Hichem Haddou; Souier, Mehdi. A Digital Twin Framework for Flexible Manufacturing System (2024) IFIP Advances in Information and Communication Technology, 733 IFIP, pp. 155 - 167, Cited 0 times. DOI: 10.1007/978-3-031-71645-4_11 https://www.scopus.com/inward/record.uri?eid=2-s2.0-85204640820&doi=10.1007%2f978-3-031-71645-4_11&partnerID=40&md5=19b2e1165b879a432ac876fb72f79767

[2] Saldarriaga, D. L. (2019). Almacenes y centros de distribución. Manual para optimizar procesos y operaciones. Marge Books. ISBN 9788417903077

[3] G. Lugaresi, N. Frigerio, and A. Matta, (2020). “A New Learning Factory Experience Exploiting LEGO For Teaching Manufacturing Systems Integration,” Procedia Manufacturing, vol. 45, pp. 271–276,

2020, https://doi.org/10.1016/j.promfg.2020.04.106.
[4] Pachacama Oña, W. C., & Salazar Muzo, J. F. (2016). Diseño y construcción de un prototipo de robot delta para aplicaciones pick & place (Bachelor's thesis, Quito, 2016). URI: http://bibdigital.epn.edu.ec/handle/15000/16931
[5] López, J., & Martínez, A. (2018). La paletización en la cadena de suministro: Técnicas y beneficios operativos. Revista de Logística y Gestión Empresarial, 34(2), 125–140.
[6] González, P., & Pérez, R. (2020). El embalaje como estrategia en la protección y presentación de productos. Revista Internacional de Ingeniería y Logística, 15(3), 58–72.
[7] Castro, E., & Hernández, V. (2019). Uso de pallets en la optimización de espacios y costos logísticos. Revista de Ingeniería Industrial y Logística, 22(4), 200–215.
[8] García, L., & Torres, F. (2020). Programación de sistemas automatizados en la industria: Aplicaciones y beneficios. Tecnología y Gestión Industrial, 12(1), 90–105
[9] Escobar Viveros, O. E., Gomez Jaramillo, M. F., & Ruiz Castillo, B. A. (2019). Construcción de un brazo robótico tipo Scara para la simulación de procesos industriales empleando piezas de lego technic: una aplicación práctica para el laboratorio de mecatrónica de la Institución Universitaria Antonio José Camacho. URI: https://repositorio.uniajc.edu.co/handle/uniajc/1120
[10] Fernández, C. (2019). Prototipado y validación de sistemas de ingeniería: Metodologías y prácticas. Revista de Diseño y Desarrollo Tecnológico, 17(2), 30–45.
[11] LEGO Education. (2016). LEGO Mindstorms EV3 Education. [PDF] LEGO® Education
[12] Sánchez, R., & Ramírez, L. (2021). Prototipado con LEGO MINDSTORMS: Innovación educativa y en robótica. Revista de Robótica Educativa, 7(1), 50–66.
[13] Toapanta Tipanta, J. C. (2014). Diseño y construcción de un prototipo para una máquina semiautomática envolvedora de palets portátil con plataforma giratoria y sistema de sujeción de film (Bachelor's thesis). URI: https://dspace.ups.edu.ec/handle/123456789/7454
[14] GitHub Inc (2024). EV3Dev: A Debian Linux-based operating system for LEGO MINDSTORMS. GitHub. November 4, 2024, from https://github.com/ev3dev/ev3dev/wiki
[15] GitHub Phyton (2025). EV3Dev: Python language bindings for ev3dev. 2025, GitHub Inc, from https://github.com/ev3dev/ev3dev-lang-python
[16] T. H. Hughes, G. H. Willetts, and J. A. Kryczka, LQG controller for the LEGO MINDSTORMS EV3 Gyroboy Segway robot, *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 17282-17287, 2020. https://doi.org/10.1016/j.ifacol.2020.12.1811