TECHNIUM

Technium Vol. 20, pp.1-15 (2024)
ISSN: 2668-778X
www.techniumscience.com

Design and development of an automated product-fetching platform for electronic orders

Konstantinos Vlassis, Evi Papaioannou, Christos Kaklamanis

University of Patras and CTI “Diophantus”, Greece
papaioan@ceid.upatras.gr

Abstract. We present "E-shopaholic," an innovative e-commerce platform that integrates advanced automation techniques to optimize the online shopping experience. "E-shopaholic" exploits automation to address challenges of traditional e-commerce systems, particularly in order-picking. Leveraging the Internet of Things (IoT), "E-shopaholic" employs a NodeMCU ESP8266 module and a servo motor to simulate the picking and placing of products, aiming to contribute to the evolution of efficient e-commerce operations through automation. The platform, with its real-time automated order processing and updates, aims to enhance operational efficiency by lowering order processing times, increasing accuracy, and improving the overall order fetching process. The system design and implementation are discussed in detail, highlighting the potential of IoT and automation to transform e-commerce practices. "E-shopaholic" serves as a proof of concept for scalable and cost-effective solutions in the e-commerce sector, suggesting a path forward for research and development in automated systems.

Keywords. E-commerce, electronic orders, Internet of Things, automated system, NodeMCU ESP8266, servo motor, order picking, WordPress.

1. Introduction

The e-commerce industry, a critical component of the global economy, has experienced exponential growth over the past few decades. This growth is propelled by the integration of advanced technologies and changing consumer preferences, leading to a transformative impact on practices for online sales. However, alongside its rapid expansion, the e-commerce sector faces substantial operational challenges, particularly regarding fetching processes. These challenges may have significant implications on efficiency, cost, and customer satisfaction.

In traditional e-commerce models, order fetching is predominantly manual. This approach, while initially straightforward, has exhibited considerable limitations, particularly as online shopping volumes increase. Customer requirements and the need for flexible processes have made the manual operations at warehouses suboptimal [8]. Manual order picking, a key stage in order fulfillment, is prone to errors and labor-intensive, as it utilizes up to 60% of the labor force [15]. It is also one of the most costly and time-consuming activities in warehouse operations. The inefficiencies inherent in this method, such as slow processing times and the potential for human error, directly impact customer satisfaction and the economic viability of e-commerce operations. In general, the method of order picking is one of the most crucial factors in the success of an e-commerce business, as it accounts for about 50-75% of the total operating cost for a typical warehouse [5]. The collecting phase in general,

is divided into four stages: travel time, search time, retrieval time and returning time. Travel time alone accounts for 27.5% of the total cost [3]. The modification and improvement of all those stages is essential in optimizing the whole collection process [4].

Automation in e-commerce product fetching provides a promising solution, successfully addressing these challenges. Automated or robotized order-picking systems offer numerous advantages over traditional methods [7, 13]. Firstly, automated order-picking systems significantly enhance efficiency and accuracy in the collection process [16]. Automated systems can process orders faster, accelerating the picking process, reducing the times of the four collection stages, which implies faster fulfillment lead time, from order placement to dispatch [6]. This speed of the overall process from order placement to dispatch is crucial in an industry where customer expectations for rapid delivery are continuously increasing. Furthermore, automation minimizes human error, which is one of the main reasons why the need for automation was essentially developed [1], leading to a more accurate order fetching in general. Enhanced efficiency and accuracy during the collection phase can, consequently, lead to higher customer satisfaction [16]. Secondly, automation offers scalability and throughput flexibility, which are essential in e-commerce environments [2]. Automated systems can provide 24/7 operation schedules and can easily adapt to changes in order volume, which is particularly beneficial during peak shopping seasons or sales. This scalability ensures that e-commerce platforms can maintain high service levels even under fluctuating demand, a flexibility that is difficult to achieve with manual processes.

Furthermore, by optimizing the order picking process, automated systems can lead to substantial long-term savings. Although the initial investment required for adopting these systems is high, with the right planning and implementation, it can be balanced out by the reduction of operation costs [14], which in the course of time can generate significant profits for the company. As the cost of labor increases over time, automation in order picking is getting more important [14]. In traditional warehouses, investments for hiring and training new employees are constantly being incurred and errors that occur due to mispicks can lead to significant losses through returns [10]. Also, the slow processing speeds regarding travel, search, retrieval and returning times are always an issue for traditional warehouses. The usage of automated systems addresses those issues, as it results in the reduction of required work force, lead-time, and errors during order fetching [13]. Due to their speed, precision, their ability to eliminate human labor and their overall efficiency in picking processes, automated systems can consequently lower ongoing labor costs in general [12].

Additionally, automated order-picking systems operate with greater energy efficiency, further reducing operational expenses [6]. Automated warehouses can function optimally with minimal lighting and climate control, leading to further reductions in facility-related costs. The ecological aspect of automated systems cannot be overlooked. By optimizing space usage and especially by minimizing lighting and heating, which are the two most energy-consuming services [9], automated order fetching centers contribute to a more sustainable and environmentally friendly operational model. This aspect aligns with the increasing global emphasis on reducing the environmental impact of business operations.

"E-shopaholic" emerges as a relevant case study within this context as it serves as a proof-of-concept platform demonstrating the application of automation in e-commerce order fetching. The platform integrates an ESP module as its central unit, which controls a servo motor to simulate the automated picking process. While it is a prototype, "E-shopaholic" represents a step towards understanding and implementing automation in e-commerce. Due to its inexpensive components, the system is intended both for small to medium-sized enterprises (SMEs) as well as large ones that are looking to enhance efficiency and reduce operational costs.

The focus of this work is to examine the potential of automated systems in improving the order fetching process in e-commerce. "E-shopaholic" is used as a model to explore improvements during the order fetching process, particularly focusing on the benefits of automating the order picking process. This study aims to provide insights into how IoT technologies can be employed to address the

inefficiencies of traditional e-commerce systems and to propose a scalable, cost-effective solution for enhancing operational efficiency.

The rest of the paper is structured as follows. After elaborating on the existing challenges in traditional e-commerce order-fetching processes, underscoring the need for innovative solutions such as automation and IoT integration (Section 1), we next present the methodology employed in the development of "E-shopaholic," outlining the system design and the technological rationale (Section 2). Then, we discuss implementation details, including challenges encountered and solutions developed (Section 3). Finally, we conclude (Section 4) by reflecting on the implications of our work, discussing its limitations, and suggesting directions for future research.

2. System design

In this section, we present design details of our proof-of-concept automated e-commerce system. The website serving as our digital storefront supporting the system is available at: http://e-shopaholic.atwebpages.com/.

2.1. System overview

"E-shopaholic" is an innovative e-commerce platform that integrates automation aiming to enhance the online shopping experience. The platform itself is designed to serve as an e-shop, posing as an online front to a broad range of businesses, aiming to streamline their online sales processes.

The platform integrates various technologies, including web hosting services and IoT technology. This integration is designed to create a smooth, user-friendly online shopping environment through a well-structured and fully functional website, as well as a coherent system that supports the automated functions of the platform.

The system is engineered to simulate the real-world automated picking process, with the intend to showcase the potential for automation within e-commerce operations. By automating key aspects of the order fulfillment process, "E-shopaholic" aims not only to improve operational efficiency but also to enhance the overall customer shopping experience. In other words, "E-shopaholic" demonstrates the potential of integrating automation in e-commerce, aiming to provide an efficient, scalable, and adaptable solution for modern online marketplaces.

2.2. User interface and functionalities

The "E-shopaholic" platform follows a clear and intuitive website design that allows users to navigate effortlessly through available product offerings and easily place orders. The homepage serves as the gateway, highlighting featured products and providing direct access to product categories through a concise menu bar. Each category page, like Kitchen, Living Room, and Office, is curated to showcase available products, enabling customers to add items to their cart seamlessly (Fig. 1).

Technium Vol. 20, pp.1-15 (2024)
ISSN: 2668-778X
www.techniumscience.com

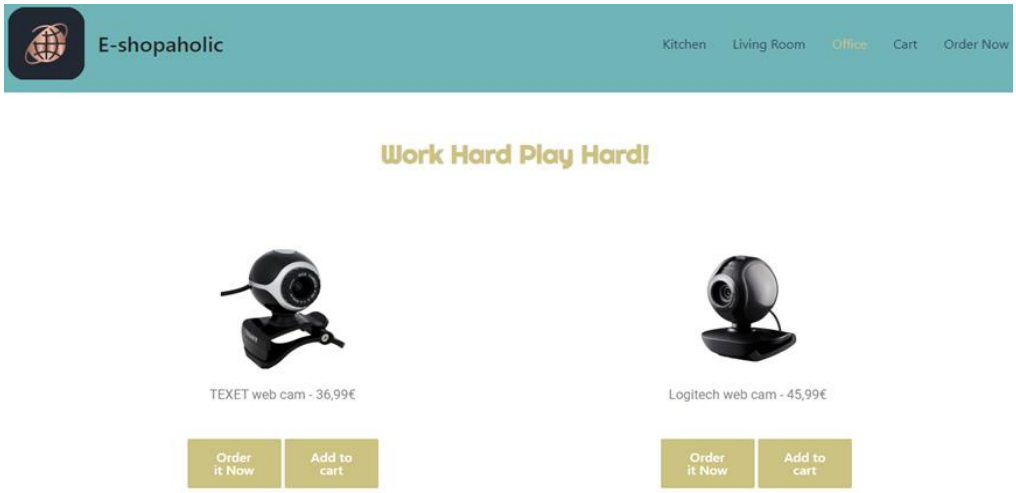


Figure 1: Product-displaying pages

The cart system is engineered for user convenience, indicating added items with a simple notification badge, which, when interacted with, triggers a pop-up displaying a summary of all selected items and their corresponding quantities as well as the total cost (Fig. 2). This design choice streamlines the shopping experience by allowing users to quickly review their multiple selections and proceed directly to the order form page. The platform also enables users to proceed immediately to the order page with their desired item by clicking on the "Order it now" button, offering even further flexibility.

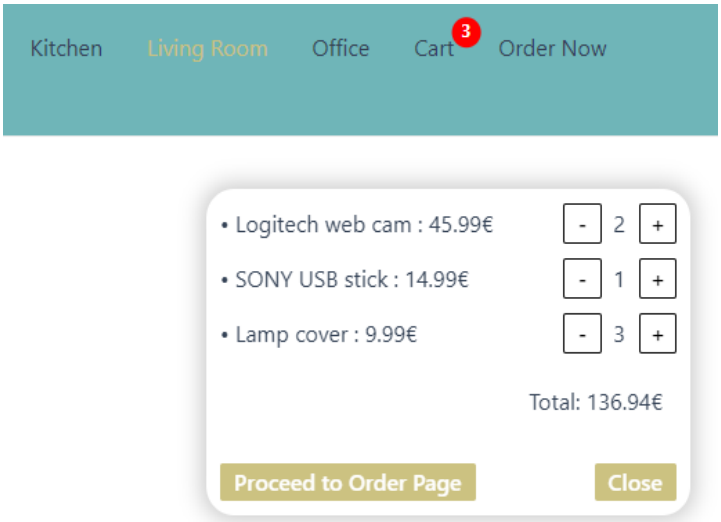


Figure 2: Cart system (pop-up)

The order form contains necessary customer details and selected products, which are then stored for order processing upon submission (Fig. 3). Users can adjust the selected items and their quantities on the order form itself, via a drop-down menu (Fig. 4), thus offered complete freedom over their item list manipulation.

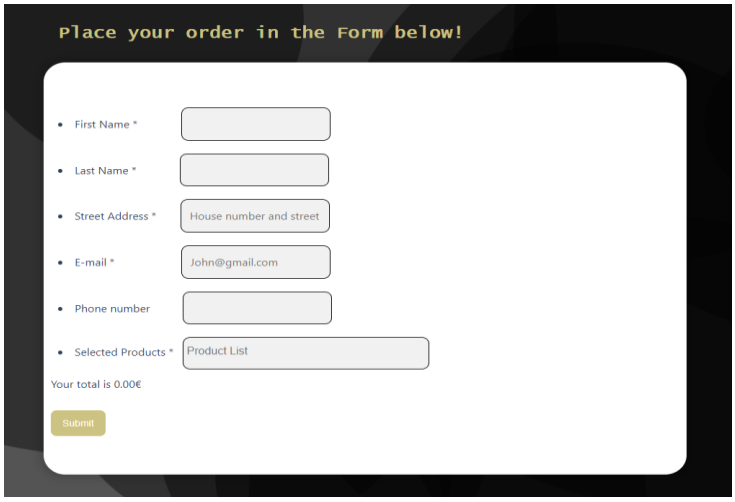


Figure 3: Order form

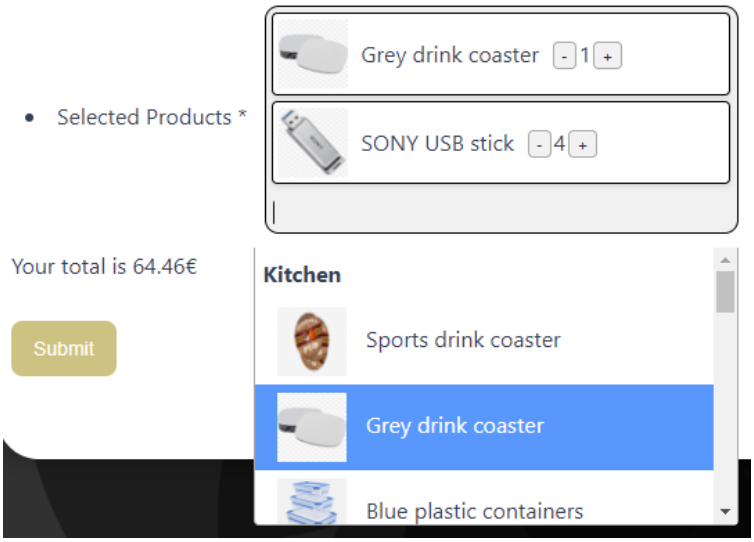
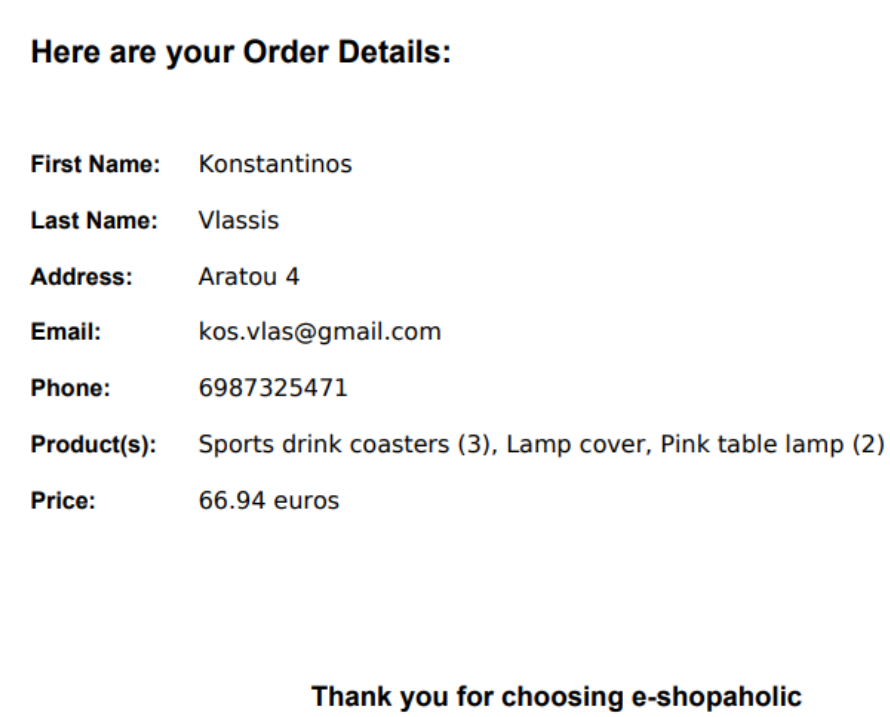


Figure 4: Drop-down menu

Upon successful order placement, users reach a confirmation page, featuring a "success" message, affirming successful order-placement, as well as a link to an automatically-generated PDF file containing detailed order information (Fig. 5). Additionally, this particular page offers real-time updates on the progress of the order through automated notifications. These notifications inform the user as each selected item is retrieved, while a final message confirms the completion of all item retrievals (Fig. 6), ensuring that users are kept informed throughout the process.



Here are your Order Details:

First Name:	Konstantinos
Last Name:	Vlassis
Address:	Aratou 4
Email:	kos.vlas@gmail.com
Phone:	6987325471
Product(s):	Sports drink coasters (3), Lamp cover, Pink table lamp (2)
Price:	66.94 euros

Thank you for choosing e-shopaholic

Figure 5: Automatically-generated PDF file with order details

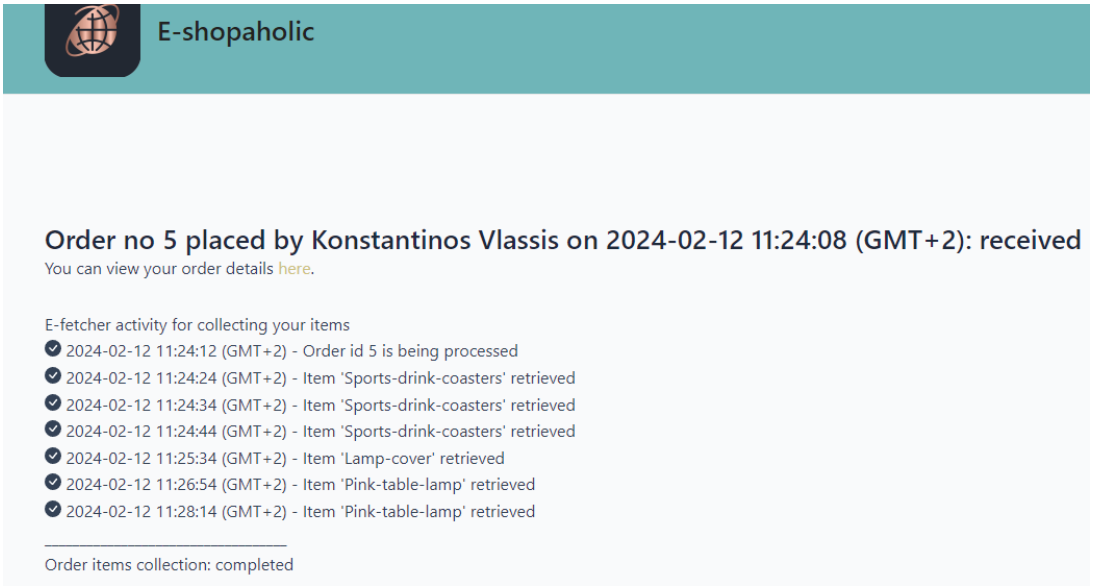


Figure 6: Order progression notifications

The integration of user interface and functionalities within "E-shopaholic" is carefully designed to ensure a user-friendly and efficient online shopping experience. Enhanced by intuitive navigation and a streamlined checkout process, the platform facilitates a smooth transition from browsing to order completion. Emphasizing user empowerment, it offers tools for order customization and real-time updates on order status, underscoring the focus of the platform on transparency and customer engagement.

2.3. System architecture

The architecture of "E-shopaholic" is meticulously crafted to provide an integrated e-commerce solution, facilitating a fluid interaction between the user interface and the automated backend processes. The core architecture merges intuitive web design with a backend automation framework, utilizing a microcontroller-WiFi module for digital-to-physical task translation and a moving mechanism for the physical handling of orders.

This dual-layer framework ensures seamless transition from user activities on the website, like navigating and placing orders, to the automated processing tasks executed by the hardware components of the system. Hardware components are essential for simulating the tangible aspects of the e-commerce experience, seamlessly connecting digital user interactions with real-world order processing.

The system architecture (Fig. 7) showcases the logical flow from user input through the processing stages, starting with data storage, followed by data retrieval via the microcontroller, progressing to the automated fetching executed by the mechanism, to finally sending feedback back to the website. This diagram serves as a visual representation of the system main components and their interconnections, illustrating the data flow and control logic that underpin the operational efficacy of the platform.

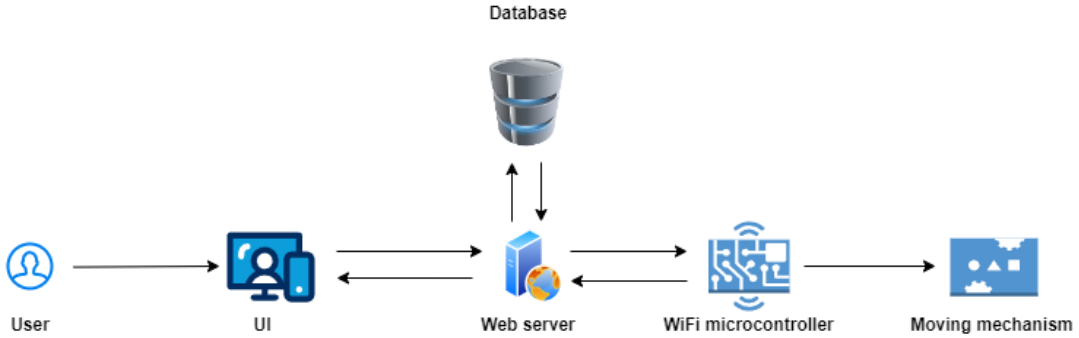


Figure 7: System architecture

By adopting this architecture, "E-shopaholic" encapsulates a dynamic approach to e-commerce and aims to offer a scalable, adaptable, and efficient solution for e-commerce platforms looking to leverage automation for improved operational performance and customer satisfaction.

3. System implementation and operation

In this section, we present implementation details. We start by reviewing the various technologies and components selected for the development of the system, as well as their cohesive integration. We focus on particular hardware components, providing more details about their role in the system. A detailed description of the servo motor operation and product positioning is also presented in this section, providing a clear demonstration of our physical system.

3.1. Technology selection

For developing "E-shopaholic," a careful selection of technologies was made to align with the platform's functional requirements and objectives. AwardSpace [18] was chosen to host the website due to its compatibility with various development platforms. That combined with its ease of use made AwardSpace our preferred choice for hosting the e-commerce platform.

The development platform of choice was WordPress [30], selected for its widespread use, robust community support, and the flexibility it offers. WordPress's extensive range of plugins and customizable themes enabled the creation of a tailored e-commerce experience. To further enhance the

frontend design, we used the WordPress plugin Elementor [19] due to its developer-friendly interface, allowing for the rapid development of an approachable and engaging user interface. Custom coding, particularly through the theme's directory, was employed to integrate specific functionalities and make “E-shopaholic” a fully-fledged functioning website. For database-management purposes we used PhpMyAdmin [27] which helps in efficient and secure handling of order data and customer information.

We chose the NodeMCU ESP8266 module [26] as the central microcontroller unit and the servo motor SG90 [29] as our fetching mechanism. The selection of these hardware components was mainly due to cost-effectiveness and functionality reasons. NodeMCU ESP8266 is a highly favored choice due to its Wi-Fi capability, compact size, and affordability, making it suitable for a variety of IoT projects [11], where wireless communication is important. The choice of using the Arduino IDE (Integrated Development Environment) [17] for programming the ESP8266 module was driven by its adaptability, extensive library support, and compatibility with various IoT devices. Arduino IDE facilitated the integration of the ESP8266 module, as well as the development of the firmware necessary for the ESP8266 to control the servo motor and maintain effective communication with the e-commerce platform. The SG90 servo motor was chosen for its precision in control and positioning. Servo motors are ideal for applications where the movement of mechanisms must be controlled with accuracy, such as in simulating the picking process of products. The ability of servo motors to maintain torque at low speeds and the ease of controlling their position through pulse-width modulation signals make them indispensable for tasks requiring reliable and controlled movement.

Together, NodeMCU ESP8266 and SG90 servo motor components provide a foundation that supports the automated interaction with the e-commerce platform, ensuring an efficient and reliable simulation of the order picking process. This combination of hardware enables the "E-shopaholic" platform to demonstrate the practical application of automation in enhancing online orders operations.

Additionally, scalability and system performance were key considerations, particularly in handling multiple orders with multiple products. The modular nature of WordPress and the efficiency of the ESP8266 module offered a scalable solution while ensuring consistent performance.

The selection and seamless integration of the aforementioned technologies were instrumental in the successful realization of "E-shopaholic." Each component was chosen not only for its individual strengths but also for how it complemented the other elements of the system, resulting in a cohesive and functional automated e-commerce platform.

3.2. Hardware setup

The physical construction of the "E-shopaholic" automated system commenced with the wiring of the ESP8266 module to the servo motor, a crucial step in bringing the automated order processing capability to life. The ESP8266 was connected to a standard servo motor, which serves as the actuator for the simulated picking process. Powering this setup, the ESP8266 was connected to a computer via USB, which also provided an interface for programming and debugging.

The servo motor's wiring was straightforward yet critical for reliable operation. The motor's voltage control pin (VCC) was connected to the ESP8266's 3.3V pin to ensure a stable power supply. The ground pin (GND) from the motor was tied to one of the ESP8266's ground ports, establishing a common ground for the circuit. The signal pin, which controls the motor's positional movements, was linked to the D7 digital pin on the ESP8266, allowing for precise control signals to be sent from the microcontroller to the servo (Fig. 8).

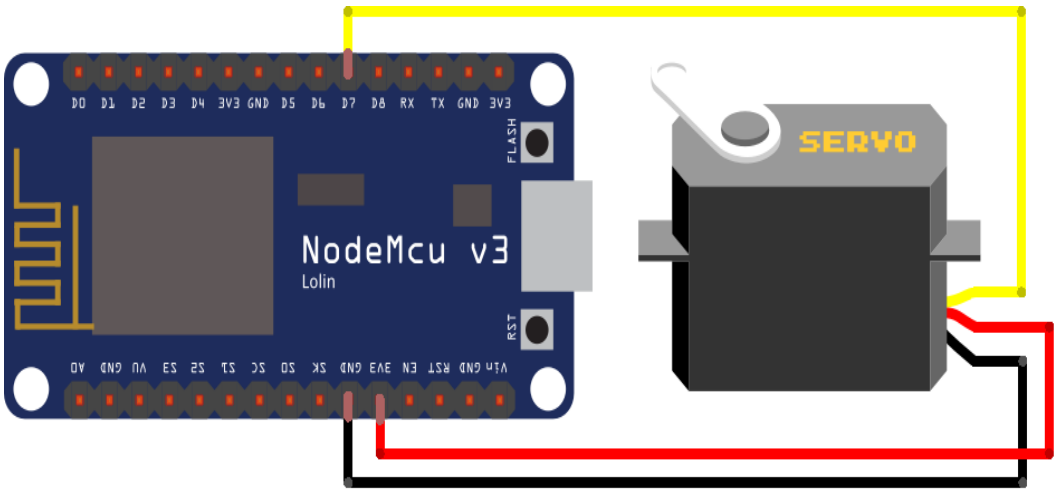


Figure 8: ESP and servo wiring

With the hardware components wired and powered, the ESP8266 was ready to be programmed with the logic required to interpret order information and drive the servo motor to the correct positions, corresponding to the virtual representation of product locations in the e-commerce platform.

This setup marked the transition from the theoretical design to the tangible operation of "E-shopaholic." It led to the subsequent phases of software configuration and development, which are detailed in the following section.

3.3. Software development and configuration

In this section, we transition from the selection of key hardware components and the assembly of the physical system to the software development that sets up the entire operation. A two-pronged approach to software configuration is presented. Firstly, we expand on the user interface and functionalities section, providing further details about the web development process. Secondly, we delve into the development of the ESP8266, detailing the programming that enables real-time communication and control over the order processing mechanism. Each aspect plays a pivotal role in realizing the "E-shopaholic" system as a fully integrated, automated e-commerce solution.

3.3.1. Web development. The web development aspect of "E-shopaholic" concentrated on essential functionalities that required custom scripts, beyond the creation of static pages and elements with Elementor. The programming languages used for the development of “E-shopaholic” were mainly CSS [21], HTML [22], JavaScript [23], and PHP [24], with SQL [25] for database interactions.

For the cart system, a “dummy” element was created in the menu bar. When clicked, instead of directing the user to another page, it opens a pop-up. When the “Add to cart” button is clicked, the name of the corresponding product and its price are stored in the local storage. The cart then fetches that data, adds them to the product list and calculates the total cost. When the quantity adjustment buttons are clicked the product list and the total cost also get adjusted accordingly.

The order form page contains a basic form structure, with the exception of the final field, named “selected products”, which is a multiple select element, created using the select2 library [28] and modified to include images and quantity adjustment buttons. When the user redirects to the order form page via the cart pop up, the local storage is cleared, and the product list is displayed in the selected products field. Upon form submission, all form data is transferred, via POST method, to the functions.php file, which is the nodal point inside the WordPress installation, since it allows communication with the WordPress database and every other component inside the installation.

Within that file there is a custom function that updates the database, adding the new order data as a new row in the "wp_orders" table. The table fields include customers personal information, their selected products, and their corresponding quantities. There is also an additional field called "executed" that can have two potential values, either "yes" or "no", signaling whether the order has been processed or not (Fig. 9).

id0	first	last	address	email	phone	product_name	quantity	executed
3	Konstantinos	Gian	Theo Vres 94	kostas.v@gmail.com	6945628526	Grey-candles,Sharp-digital-clock,Ideus-handsfree	2,1,3	yes
4	Ntinos	V	Riga 23	v.ntinos@outlook.com	6975429503	Grey-drink-coasters	1	yes
5	Konstantinos	Vlassis	Aratou 4	kos.vlas@gmail.com	6987325471	Sports-drink-coasters,Lamp-cover,Pink-table-lamp	3,1,2	yes

Figure 9: Database

Upon successful database entry, another added function within the functions.php file creates the information PDF file using the FPDF library [20] and redirects the user to the confirmation page, with the unique order id and the number of products ordered, as URL parameters. Order id is used to create a unique order tracking page for the user and the number of products is used so that the server knows how many product updates it will receive from the ESP8266.

A pivotal aspect of the development phase was the establishment of two endpoints, which facilitate the real-time synchronization between the website and the microcontroller's processing sequence. The first endpoint was created via a custom WordPress plugin and was set up as an access point for the ESP8266, enabling it to fetch the order data from the website. Since the microcontroller cannot communicate with the database directly, WordPress API was used to transfer the data stored in the database to the endpoint in JSON format (Fig. 10), a crucial step for initiating the automated order processing sequence. The second endpoint was created via a custom script and was designed to handle requests and receive data, in the form of messages, from the ESP8266, effectively communicating the status of order processing back to the website for real-time user updates. The website, upon receiving the ESP8266's confirmations, updates the order status and immediately reflects these changes on the user's confirmation page. Users can watch the progression of their order in real-time, from processing to the simulated picking of their products, all the way to the order's completion.


```
[{"id0":"3","first":"Konstantinos","last":"Gian","address":"Theo Vres
94","email":"kostas.v@gmail.com","phone":"6945628526","product_name":"Grey-candles,Sharp-digital-clock,Ideus-
handsfree","quantity":"2,1,3","executed":"yes"},{"id0":"4","first":"Ntinos","last":"V","address":"Riga
23","email":"v.ntinos@outlook.com","phone":"6975429503","product_name":"Grey-drink-
coasters","quantity":"1","executed":"yes"},{"id0":"5","first":"Konstantinos","last":"Vlassis","address":"Aratou
4","email":"kos.vlas@gmail.com","phone":"6987325471","product_name":"Sports-drink-coasters,Lamp-cover,Pink-
table-lamp","quantity":"3,1,2","executed":"yes"}]
```


Figure 10: Endpoint content

The integration of these endpoints represented an important advancement in creating a seamless bridge between the web platform and the physical automation components. This approach ensured that the platform could offer an interactive and responsive shopping experience.

3.3.2. ESP8266 configuration. After establishing the hardware set-up and with our website foundation in place, the focus shifted to configuring the software of the components that would manage the automated fetching process. In the programming phase of the ESP8266 module, the main priority was the establishment of three major functionalities: retrieving order data from the website, controlling the servo motor to simulate the order picking process and sending updates back to the website for each product retrieval.

The development began with writing the firmware for the microcontroller. The ESP8266 module was configured to automatically connect to the internet via Wi-Fi, utilizing pre-determined credentials.

This autonomous connectivity is essential for the system to retrieve order data without manual intervention, ensuring a seamless flow from order placement to processing. The firmware is structured to regularly poll the API endpoint for data entries that have not been processed yet. Once a new order is detected, the ESP8266 parses the JSON data to extract the order details, including the product identifiers and quantities. It then converts those strings into arrays and triggers the servo motor to move in a loop to designated positions that represent the physical locations of products in a real-world scenario. The servo's movements simulate the picking process, where, in a fully automated system, products would be retrieved from the shelves and placed on a platform marking them ready for shipping. The servo moves twice for each selected product. First, it moves to that product's corresponding position, retrieving it, then it moves back to its initial position, placing the product successfully. To enable the two-way communication necessary for real-time order status updates, the firmware also included a function to send messages back to the website through the second endpoint. Once the simulated picking of a product is completed and the servo is set on its initial position and ready for its next move, the ESP8266 sends the corresponding confirmation message to the website. That process occurs once for each product, until every product on the list has been retrieved.

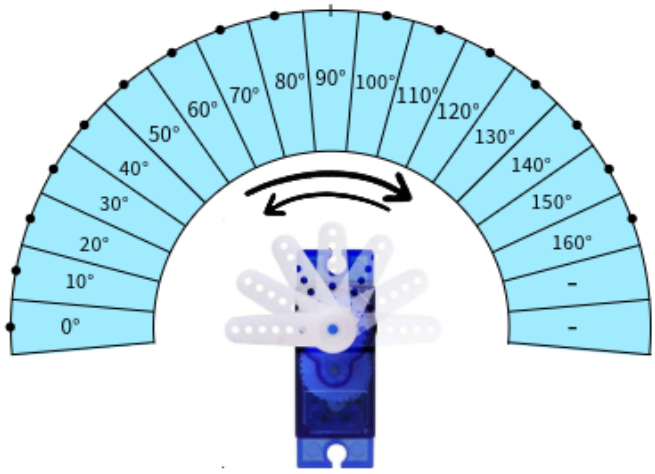
The development process was iterative, with frequent testing to ensure each component functioned as intended. Challenges such as ensuring data consistency and maintaining a stable Wi-Fi connection were addressed through a combination of rigorous testing and firmware optimization.

As the software components came together, they formed a cohesive system capable of autonomously processing orders and providing feedback to both the customers and the website administrators.

3.4. Servo motor operation and product positioning

Within "E-shopaholic," the servo motor plays a pivotal role in simulating the product fetching process. A servo motor is a rotary actuator that allows for precise control of angular position, which is essential for accurately simulating the picking of different products. In our system, the servo motor is programmed to rotate to specific angles corresponding to the virtual locations of products. The servo motor moves clockwise to retrieve the products and then moves counterclockwise to bring them back and place them on the completion platform.

"E-shopaholic" features 16 different products, each assigned to a unique position within the servo's rotational range. This range is divided into 10-degree increments, representing different sectors for each product. For example, Product 1 is located at 10 degrees, Product 2 at 20 degrees, continuing in this pattern up to Product 16 at 160 degrees (Fig. 11).



Item	Position
Placing platform	0°
"Sports drink coasters"	10°
"Grey drink coasters"	20°
"Blue plastic containers"	30°
"Orange plastic containers"	40°
"Lamp cover"	50°
"Grey candles"	60°
"Mushroom lamp"	70°
"Pink table lamp"	80°
"Sharp digital clock"	90°
"Black digital clock"	100°
"Texet web cam"	110°
"Logitech web cam"	120°
"Sony usb stick"	130°
"Cruzer usb stick"	140°
"Ideus handsfree"	150°
"Black handsfree"	160°

Figure 11: Position visualization

The timing mechanism of the servo motor is designed to reflect a realistic process of picking a product. To retrieve Product 1 at 10 degrees, the servo takes 5 seconds to reach the position and another 5 seconds to return to its starting point. As the products are located further along the servo's path, the time taken to reach them increases proportionally. Therefore, to reach Product 2 at 20 degrees, it takes 10 seconds, and similarly, 10 seconds to return. Table 1 delves deeper into that, demonstrating the picking time pattern.

Table 1. Product-picking times

Product Number	Servo Position (Degrees)	Time to Reach Position (Seconds)	Time to Return (Seconds)	Total Time (Seconds)
Product 1	10	5	5	10
Product 2	20	10	10	20
Product 3	30	15	15	30
…	…	…	…	…
Product 16	160	80	80	160

This precise control and timing of the servo motor's movements ensure a realistic and efficient simulation of the order picking process.

4. Conclusion

This analysis of the 'E-shopaholic' platform explores the integration of IoT and automation within the e-commerce sector. The development highlights the system's approach to enhancing the online shopping experience and operational efficiencies through technological advancements. The initiative demonstrates the practicality and feasibility of automating the order fetching process alongside providing real-time consumer updates. However, it acknowledges constraints, such as the need for empirical validation and the challenges of scalability in real-world diverse market environments with diverse products and storages. As a proof of concept, 'E-shopaholic' suggests additional investigation to refine automation technologies in e-commerce logistics. Future work could focus on addressing these challenges, improving system adaptability, and evaluating the impact on consumer satisfaction and business performance. This study thus contributes to the ongoing discourse on the potential of automated systems to enhance e-commerce practices, challenging the high-cost industry norms and encouraging further research and development in this evolving field.

References

[1] A. M. Atieh et al., “Performance improvement of inventory management system processes by an automated warehouse management system,” Procedia CIRP, vol.41, no.1, pp.568–572, 2016, doi: https://doi.org/10.1016/j.procir.2015.12.122.

~~[3]~~[2] K. Azadeh, R. De Koster, and D. Roy, “Robotized and Automated Warehouse Systems: Review and Recent Developments,” Transportation Science, vol. 53, no. 4, pp. 917–945, Jul. 2019, doi: https://doi.org/10.1287/trsc.2018.0873.

~~[5]~~[3] C. Bitterling, J. Koreis, D. Loske, and M. Klumpp, “Comparing manual and automated production and picking systems,” www.econstor.eu, 2022. http://hdl.handle.net/10419/267191.

~~[7]~~[4] G. P. Broulias, E. C. Marcoulaki, G. P. Chondrocoukis, and L. G. Laios, “Warehouse Management for Improved Order Picking Performance: An Application Case Study from the Wood Industry,” Jan. 2005.

~~[9]~~[5] J. J. Coyle, E. J. Bardi, and C J. Langley, The Management of Business Logistics: A Supply Chain Perspective, 7th edition, St. Paul, South-Western College Pub, 2002.

~~[11]~~[6] A. Dhaliwal, “The Rise of Automation and Robotics in Warehouse Management,” Transforming Management Using Artificial Intelligence Techniques, pp. 63–72, Nov. 2020, doi: https://doi.org/10.1201/9781003032410-5.

~~[13]~~[7] S. Kumar, J.-B. Sheu, and T. Kundu, “Planning a parts-to-picker order picking system with consideration of the impact of perceived workload,” Transportation Research Part E: Logistics and Transportation Review, vol. 173, p. 103088, May 2023, doi: https://doi.org/10.1016/j.tre.2023.103088.

~~[15]~~[8] C. K. M. Lee, Y. Lv, K. K. H. Ng, W. Ho, and K. L. Choy, “Design and application of Internet of things-based warehouse management system for smart logistics,” International Journal of Production Research, vol. 56, no. 8, pp. 2753–2768, Oct. 2017, doi: https://doi.org/10.1080/00207543.2017.1394592.

~~[17]~~[9] K. Lewczuk, M. Kłodawski, and P. Gepner, “Energy Consumption in a Distributional Warehouse: A Practical Case Study for Different Warehouse Technologies,” Energies, vol. 14, no. 9, p. 2709, May 2021, doi: https://doi.org/10.3390/en14092709.

~~[19]~~[10] A. Miller, "Order Picking for the 21st Century: Voice vs. Scanning Technology," Tompkins Associates, 2004.

~~[21]~~[11] A. Naladkar et al., "IOT-BASED HOME AUTOMATION USING TELEGRAM USING ESP 8266," International Research Journal of Modernization in Engineering Technology and Science, vol. 5, no. 12, pp. 306-310, Dec. 2023, doi: https://www.doi.org/10.56726/IRJMETS47011.

~~[23]~~[12] A. R. F. Pinto, M. S. Nagano, and E. Boz, "A classification approach to order picking systems and policies: Integrating automation and optimization for future research," Results in Control and Optimization, vol. 12, p. 100281, Sep. 2023, doi: https://doi.org/10.1016/j.rico.2023.100281.

~~[25]~~[13] M. Urzúa, A. Mendoza, and A. González, "EVALUATING THE IMPACT OF ORDER PICKING STRATEGIES ON THE ORDER FULFILMENT TIME: A SIMULATION STUDY," Acta logistica, vol. 6, no. 4, pp. 103–114, Dec. 2019, doi: https://doi.org/10.22306/al.v6i4.129.

~~[27]~~[14] J. Varghese and S. Saju, "Challenges while moving towards Warehouse Automation," Dissertation, 2021.

~~[29]~~[15] J. Won and S. Olafsson, "Joint order batching and order picking in warehouse operations," International Journal of Production Research, vol. 43, no. 7, pp. 1427–1442, Apr. 2005, doi: https://doi.org/10.1080/00207540410001733896.

~~[31]~~[16] N. Ylä-Autio, "Optimizing warehouse operations using automation and artificial intelligence," Aalto University, 2021.

~~[33]~~[17] Arduino IDE. url: https://docs.arduino.cc/software/ide-v1/tutorials/Environment/.

~~[35]~~[18] AwardSpace. url: https://www.awardspace.com/

~~[37]~~[19] Elementor. url: https://elementor.com/.

~~[39]~~[20] FPDF library. url: http://www.fpdf.org/

~~[41]~~[21] mdn. CSS: Cascading Style Sheets. url: https://developer.mozilla.org/en-US/docs/Web/CSS.

~~[43]~~[22] mdn. HTML: HyperText Markup Language. url: https://developer.mozilla.org/en-US/docs/Web/HTML.

~~[45]~~[23] mdn. JavaScript. url: https://developer.mozilla.org/en-US/docs/Web/JavaScript.

~~[47]~~[24] mdn. PHP: HyperText Preprocessor. url: https://developer.mozilla.org/en-US/docs/Glossary/PHP.

~~[49]~~[25] mdn. SQL: Structured Query Language. url: https://developer.mozilla.org/en-US/docs/Glossary/SQL.

~~[51]~~[26] NodeMCU ESP8266. Detailed review. url: https://www.make-it.ca/nodemcu-details-

specifications.

~~[53]~~[27] PhpMyAdmin. url: https://www.phpmyadmin.net/.

~~[55]~~[28] Select2. url: https://select2.org/

~~[57]~~[29] Servo motor SG90. url: https://components101.com/motors/servo-motor-basics-pinout-datasheet.

~~[59]~~[30] WordPress. url: https://wordpress.org/documentation/.