Research Paper
Journal of Smart Systems Research (JOINSSR) 4(2),140-159,2023
Received: 06-November-2023 Accepted: 04-December-2023
Doi: 10.58769/joinssr.1385906

Indoor Cargo Vending System Development and Development Environment Design

Süleyman UZUN[1,3*], Mustafa Zahid YILDIZ[2,3], Tolga SUGETİREN[4], İsmail YÜKSEL[5]

[1] Computer Engineering, Faculty of Technology, Sakarya University of Applied Sacience, Turkiye, suleymanuzun@subu.edu.tr
[2] Electrical Electronics Engineering, Faculty of Technology, Sakarya University of Applied Sacience, Turkiye, mustafayildiz@subu.edu.tr
[3] ArGe Center, Canovate Electronics Industry and Trade Inc., Turkiye
[4] ArGe Center, Canovate Electronics Industry and Trade Inc., Turkiye, tolga.sugetiren@canovate.com
[5] ArGe Center, Canovate Electronics Industry and Trade Inc., Turkiye, ismail.yuksel@canovate.com

ABSTRACT

The rapid expansion of e-commerce caused by the Covid-19 pandemic has increased the significance of the logistics industry. Processes for delivery and return need to be more adaptable and effective in this situation. Smart outdoor cargo vending systems have been created to satisfy this need. These systems do, however, have some drawbacks, such as security and energy usage. In order to address the issues with outdoor vending, this paper presents an indoor smart cargo vending system with high security and minimal energy consumption. Customers have the choice to pick up or return their orders with the help of this system. Additionally, this creative solution aids e-commerce businesses in decreasing logistics expenses and raising customer satisfaction. For manufacturers of vending machines and e-commerce businesses, this proposed indoor smart cargo vending system provides solutions for all scenarios of cargo order processes. This solution combines the design of the system's hardware and software with a methodology that has been discussed in the scientific literature. The study's conclusion suggests a novel indoor smart cargo vending system that was created in response to the logistical difficulties brought on by the expansion of e-commerce. This system can have a positive impact in key areas like security, energy efficiency, and customer satisfaction, and it can be seen as a significant advancement for the logistics sector.

Keywords: Indoor Cargo Vending, System design, Development Environment Design, System software.

1 Introduction

The acceleration of e-commerce and the rise of mobility trends in today's rapidly digitalizing world have fundamentally altered consumer behaviors and business models. The significance and role of the logistics sector have been revised in light of this change. Particularly, the rise of e-commerce has highlighted the necessity of prompt and reliable product delivery to customers, complicating and undermining logistics operations. E-commerce has increased the need for efficient management of processes like physical transportation, storage, delivery, and return of goods. Customer satisfaction, operational effectiveness,

* Corresponding Author's email: suleymanuzun@subu.edu.tr

and cost-effectiveness are all critically impacted by these processes. Traditional logistics techniques might not be adequate to meet these new demands, though. To address these issues, intelligent cargo vending machines are being developed [1,2]. By reducing the time and space requirements of delivery and return processes, which is one of the key logistics services, and giving online shoppers access to their orders around-the-clock, these systems significantly enhance the customer experience [3]. When the literature is examined, however, effective cargo distribution models [4,5], mathematical methods used in the location selection of vending machines [1], and mobile application-based local cargo vending machine solutions [6,7] have been prioritized rather than the development of hardware and software solutions for cargo vending machines. Some studies on cargo systems have been conducted in the literature. Karabulut et al. [1], developed a mathematical model for the cargo vending machine location selection problem. They applied their proposed model to the placement of five cargo vending machines in Buca district of Izmir province. Yaman and his friend [5], with their proposed system, aimed to increase the speed, capacity and efficiency of the currently used cargo management systems. In their proposed system, they used UHF-RFID labels instead of conventional barcode or QR code labels. In the system, they installed UHF-RFID reader antennas in cargo shipping branches, transportation vehicles and transfer centers. In this way, all cargoes are read automatically, quickly and easily. Çakır and his friend [7], tracked cargo packages using RFID technology. In this way, the separation of cargo packages from a central point was performed automatically over the tape. A study similar to the system proposed in this study was not found in the literature. Due to the growth of e-commerce and the effects of mobility trends on logistics operations, new and creative solutions are now required.

To address the issues faced by e-commerce businesses and their customers, a Smart Indoor Cargo Vending System has been developed in this study. For the first time, a comprehensive approach is used in this study to discuss the software and physical security design of the designed smart indoor vending machine, energy efficiency, continuous monitoring, design of an alarm system for system failures, management of multiple vending machines and system updates from a single center, and the design of systems that ensure the complete and accurate transfer of data flow.

2 Material and Methods

2.1 System Design

The proper and systematic management of the processes for product delivery and return in e-commerce systems is proposed. The two main sections of this system are the hardware and software designs. The emphasis in this study is more on software processes. In order for multiple software developers to work on the system's software processes at once, the environment and its features are also highlighted.

2.1.1 Establishing a Software Development Environment

Software for automated indoor cargo tracking is created using a server-client model and micro-service architecture. Utilizing the information gleaned from the analysis and field audits of the software, analyses of indoor cargo tracking automation are prepared. The prepared analyses are used to divide the work into sections. Each piece of work is entered into the Jira software as a task to be assigned to various software developers. Within the time allotted to them, each software developer completes the tasks given to them and uploads them to the system. The flow chart in Figure 1 depicts the setting and procedures set up for the creation of indoor cargo tracking automation software.

The Bitbucket software hosts all written lines of code. Over time, it is necessary to update every piece of produced code. The versions of each piece of code must be preserved for this reason. This task makes use of the git software. With the help of the comprehensive Jenkins application, the deployment procedures

for developed and programmed software projects are managed. Through the git version control system, changes made to each task are sent to the Bitbucket application. When the code is committed using Bitbucket, Jenkins starts the deployment process, which is then completed by deploying the necessary environments (develop, stage, and prod) using a Kubernetes cluster.

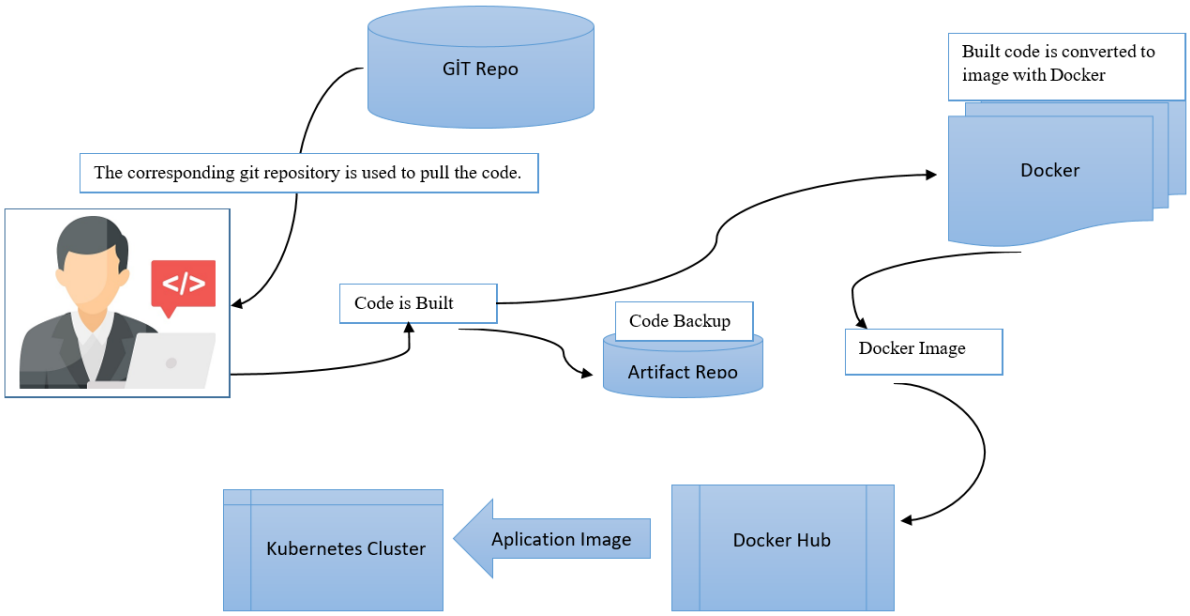

Figure 1: Schematic of the environment prepared for the development of indoor cargo tracking automation software in general

2.1.2 Establishing a Software Development Environment

Jira software is a project and business process management software developed by Atlassian [8]. It is widely used especially in areas such as software development and information technology service management [9]. This application was used during the development of the indoor cargo vending system software. In this application, all personnel involved in the project such as software developers, designers, analyzers, hardware designers and electronic card developers are involved. All processes such as work orders, completed works, etc. are managed through this application.

2.1.3 Bitbucket Application

Bitbucket is a code management and version control application developed by Atlassian to facilitate software development processes and support collaboration [10]. The codes and versions of the developed system are tracked through this software. All codes produced during the software development phase of the indoor cargo vending system are stored on this application. It is an application that can also look at old versions of the developed codes when necessary. It also supports applications such as git and mercurial for version tracking. The bitbucket application used in this study works in integration with the git version tracking application. Continuous Integration/ Continuous Delivery (CI/CD) DevOps tools have a flexible structure thanks to their integration with application monitoring and communication tools.

2.1.4 Git: Version Control Application

Git is an open source software for version control in software development projects and allows many developers to develop code in a project at the same time [11]. It was developed by Linus Torvalds in 2005 [12]. Git is distributed and each developer can create and work on their own copies. This facilitates parallel development and collaboration. It is used together with bitbucket in the established software development environment.

2.1.5 Git: Version Control Application

Jenkins is an open source automation server used as a CI/CD product to automate and manage software processes. It was first launched in 2004 as part of the Hudson project, later renamed Jenkins [13]. In the system development environment, an end-to-end deployment pipe structure was built and used to automate deployment/testing processes. The process is triggered by committing the relevant project code in the Git repo and ends with the deployment of the prepared docker image on the kubernetes cluster. The project owner is notified according to the result of the process. Software security control in the pipeline is done with SonarQube software.

2.1.6 Kubernetes Application

Kubernetes is a container orchestration platform for automating and managing container-based applications, defined as portable packages used to isolate and make portable applications and their dependencies [14]. Kubernetes organizes, controls and manages independent containers across multiple servers or cloud environments. In the software development environment designed in this study, it is installed and used on one cluster. The deployments of the developed software and services are made on these clusters. For the production environment, it is set up as 3 Master + 5 worker nodes. It can be scaled horizontally and vertically if needed. Due to the cluster structure, services are redundant and can be replicated up to the desired number of replicas.

2.2 System Hardware Design

The hardware components of the indoor cargo vending system are designed in two different ways: lockers and main unit. The sizes of the cabinets are designed in three different ways as shown in Figure 2 small, medium and large.

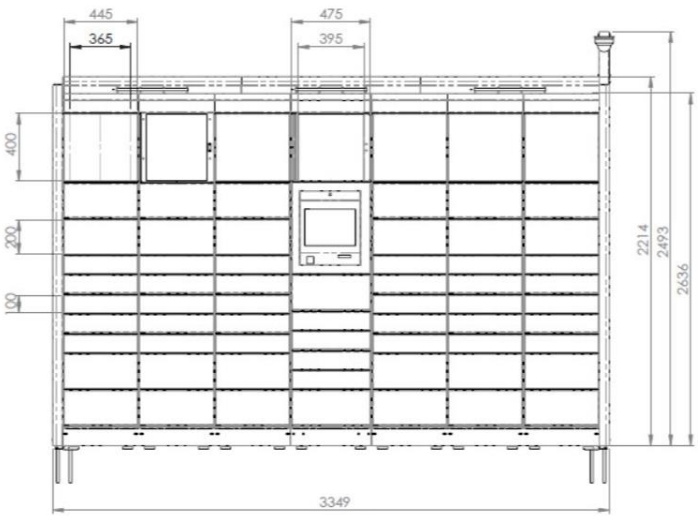

Figure 2: Sizes of the designed cargo vending machine

The front view of the prototyped lockers is shown in Figure 2. The cabinet unit consists of chambers of different sizes in which cargoes of different sizes can be placed. Since the cabinets will be stored in safer places indoors, thin sheets are used in the manufacture of the cabinet unit. Since it does not require sealing, no welding process was performed on the unit. In addition, the cabinet unit is designed to be easily assembled after cutting, bending and punching.

Figure 3: Cabinet unit of indoor cargo vending system

In addition, a camera system was added in order to take and store the images of the people making transactions. A single control card was used to control the lock mechanisms of the lockers. One computer was used to receive the cargo of the users and to operate the lock mechanism. One modem was used to provide internet connection. In the main unit;

- The case of the computer used in the system,
- Modem where the internet connection is provided,
- Network Video Recorder (NVR) device where camera recordings are saved,
- Uninterruptible Power Supply (UPS) devices used to protect devices from instantaneous power outages and sudden fluctuations in the grid,
- A smart Power Distribution Unit (PDU) that can remotely monitor power outages and how much electricity is used by connected devices,
- Lock mechanism control cards,

are available.

2.3 Indoor Cargo Vending System

The designed indoor cargo vending systems are cabinets in which e-commerce companies place their shipments, receive returns and customers receive their orders. In this way, many problems such as the customer not being found at the address for e-commerce companies, sending the cargo back to the e-

commerce company and incurring additional shipping costs have been prevented. At the same time, many problems such as not receiving their cargo in customers, adjusting themselves to the time the cargo will be delivered, have been prevented. Thanks to the cargo vending system software designed in short items;

- The Cargo personnel brings cargo and places it in the vending machine,
- Informing the e-commerce company as a result of placing the shipments in the vending machine,
- Receipt of shipments by the customer,
    - Single product pickup
    - Multiple product pickup
- Product return process,
- Product cancellation process,
- Personnel service operations

It has the opportunity to make transactions.

2.4 Indoor Cargo Vending System Screen Designs

At this stage, the screens designed on cargo vending machines and the capabilities of these screens are explained. Difficulties encountered with these screens and the measures taken are also mentioned.

2.4.1 Placing the shipment in the cargo vending machine

After the customer completes the purchase from the e-commerce company, a reconciliation number is created for the cargo personnel to place the customer's order in the cargo vending machine. After the cargo personnel place the customer's order in the cargo vending machine, a single-use password is created for the customer. This password is sent to the relevant customer as Short Message Service (SMS). The process of cargo personnel placing orders into the cargo vending machine is shown in Figure 4.

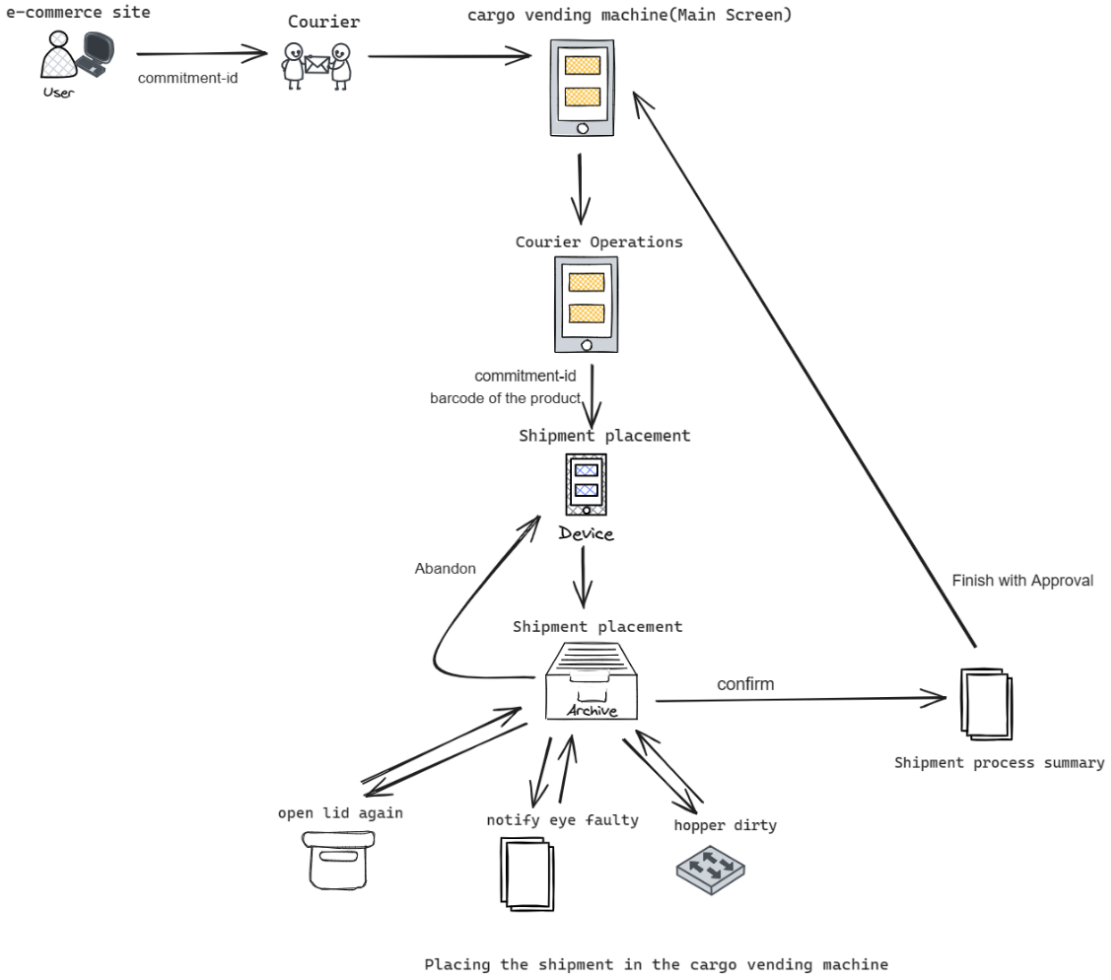

Figure 4: The process of cargo personnel placing orders into the cargo vending machine

When the cargo personnel arrive at the relevant cargo vending machine, they click on the " *Cargo personnel Operations*" menu shown in Figure 5 and enter the reconciliation number on the screen shown in Figure 6.

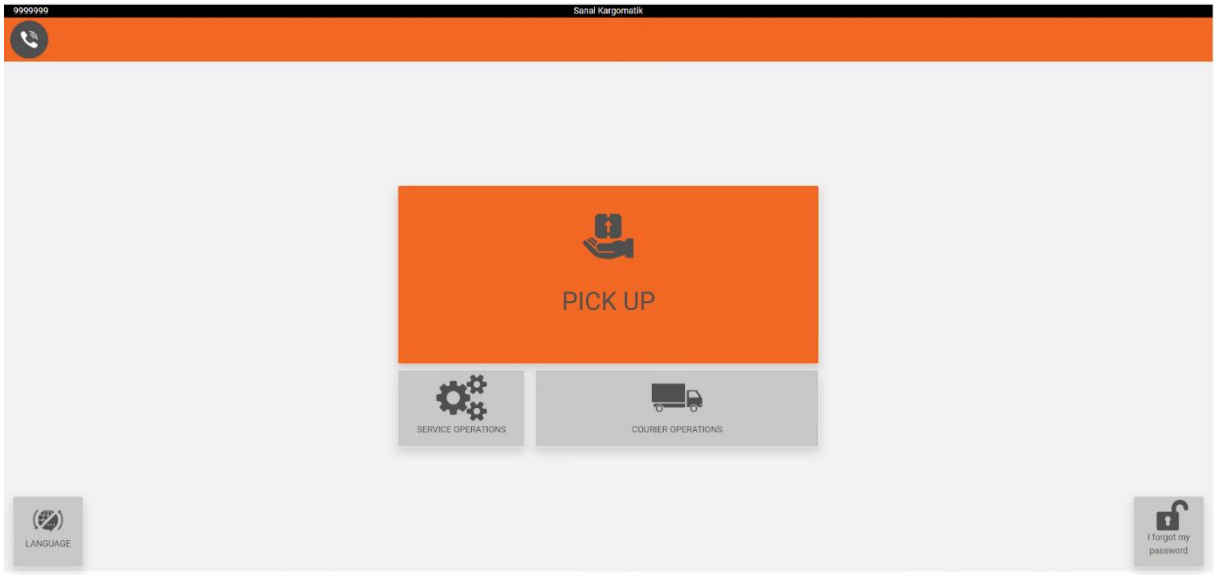

Figure 5: Cargo vending machine main screen

In Figure 5, (a) shows the unique ID of the cargo vending machine and (b) shows the name of the cargo vending machine. On the same screen, multi-language options and the ability to create a new password for those who have forgotten their password are also available.

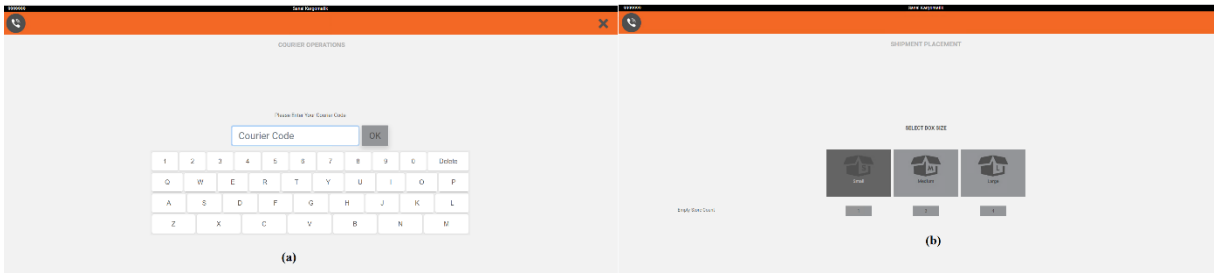

Figure 6: Product placement screens for cargo vending machines. (a) Cargo personnel operations reconciliation number entry screen, (b) Shipment placement screen.

After entering the consensus number on the screen shown in Figure 6.a, the cargo personnel can scan the barcode of the product to be placed inside the cargo vending machine or write it manually. On the screen shown in Figure 6.b, the size of the relevant cargo is selected and placed in a suitable hopper. The full or empty containers are detected by sensors and can be monitored on the screen shown in Figure 7.

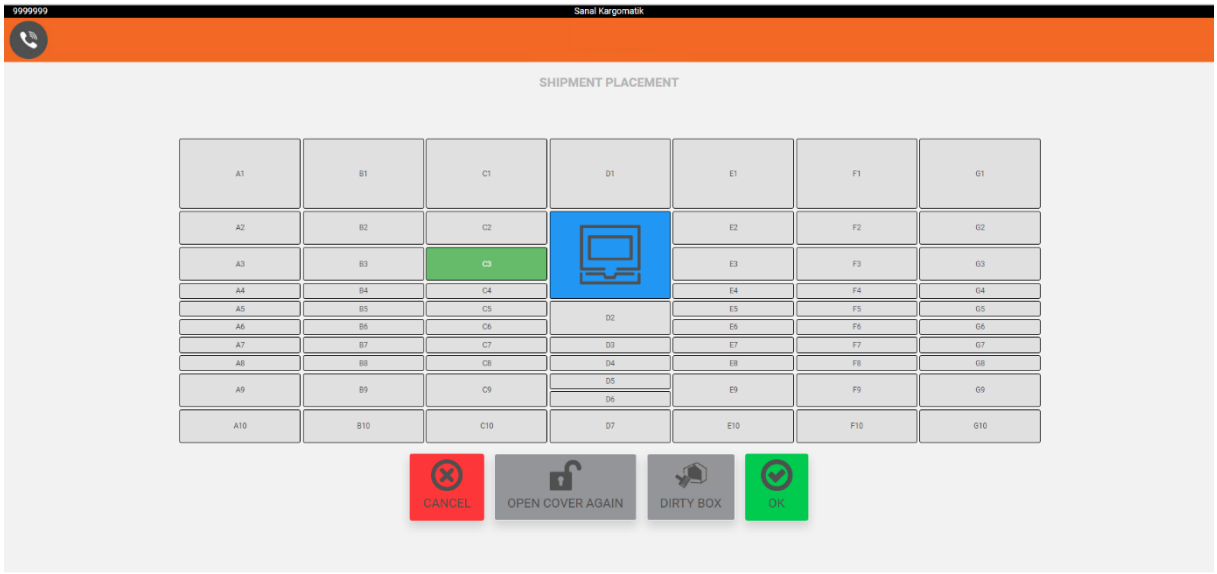

Figure 7: Shipment placement screen

After the hopper selection is made, the relevant hopper will be opened automatically. In addition, an audible warning will be made to the cargo personnel as "Please place the shipment". The "*ABANDON*" button on this screen is used in case the cargo personnel want to select a different size hopper. The "*OPEN LID AGAIN*" button is used to retrigger the lid in cases where it is not opened. If the lid opening problem persists after clicking this button, the "*OPEN LID AGAIN*" button turns into the "*NOTIFY EYE FAULTY*" button as shown in Figure 8 to report the relevant hopper as faulty and the cargo personnel reports the relevant hopper as faulty. Defective hopper notifications are monitored instantly from the portal system. Defective hoppers are disabled and reduced from the capacity on the size selection screen. The "*HOPPER DIRTY*" button has been added for the cargo personnel to use in case the hopper is dirty during the placement of the shipment. When the cargo personnel press this button, the relevant hopper will be marked as dirty and the relevant hopper will be taken out of use and reduced from the total capacity as in the faulty hopper notification process step. The "*OPEN LID AGAIN*" and "*READY TO OPEN*" buttons have

been added as measures to prevent the cargo personnel from returning to the very beginning of the shipment placement work process. In this way, it has been observed that it is very important in terms of both time and practical progress of the work.

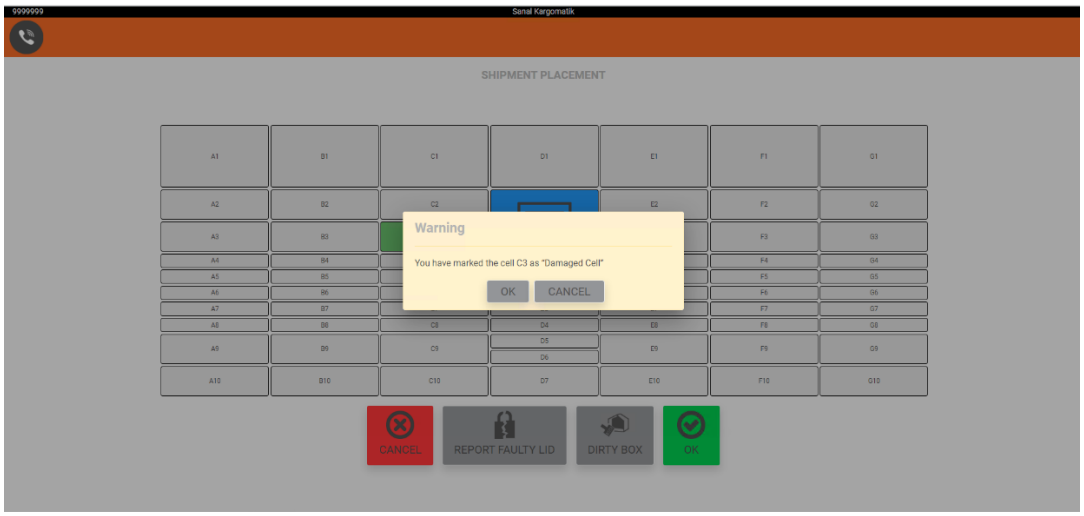

Figure 8: Hopper malfunction notification screen

The "*CONFIRM*" button is used by the cargo personnel to complete the transactions. After the cargo personnel clicks this button, the shipment transaction summary is shown as shown in Figure 9. By pressing the "*FINISH WITH APPROVAL*" button, the placement process is completed and the main screen is returned.

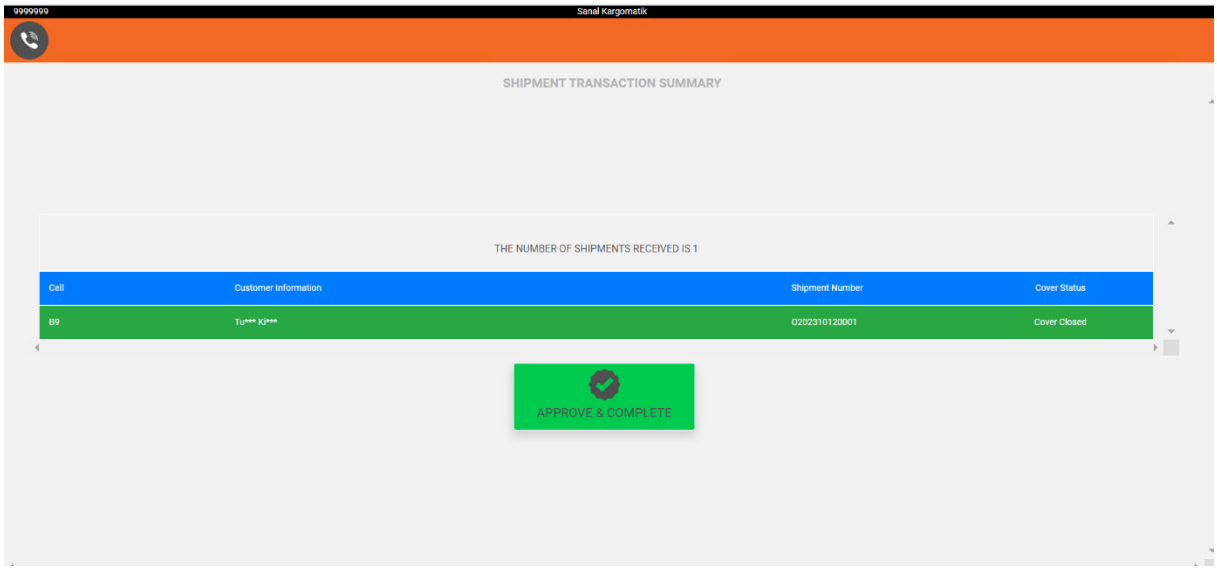

Figure 9: Shipment process summary screen

On this summary screen;

- ***Hopper information***; Information on which screen the cargo personnel placed the product,
- ***Customer information***; The name and surname of the customer for which customer the product belongs to,
- ***Delivery Number***; The order number of the shipment,

- ***Lid Status***; The closed status of the lid of the placed hopper is given. (The lid closure status control is provided by the information received from the lock switches in each hopper in cargo vending machines).

In multiple product placement operations, all operations are performed as described above. Only after the cargo personnel enters the reconciliation number, they are informed that they will enter more than one product and the personnel can complete the placement process by scanning the barcodes of all products.

2.4.2 Receiving the order from the cargo vending machine

Customers can pick up their products 24/7 from indoor cargo vending machines. The process of customers receiving their orders from the cargo vending machine is shown in Figure 10.

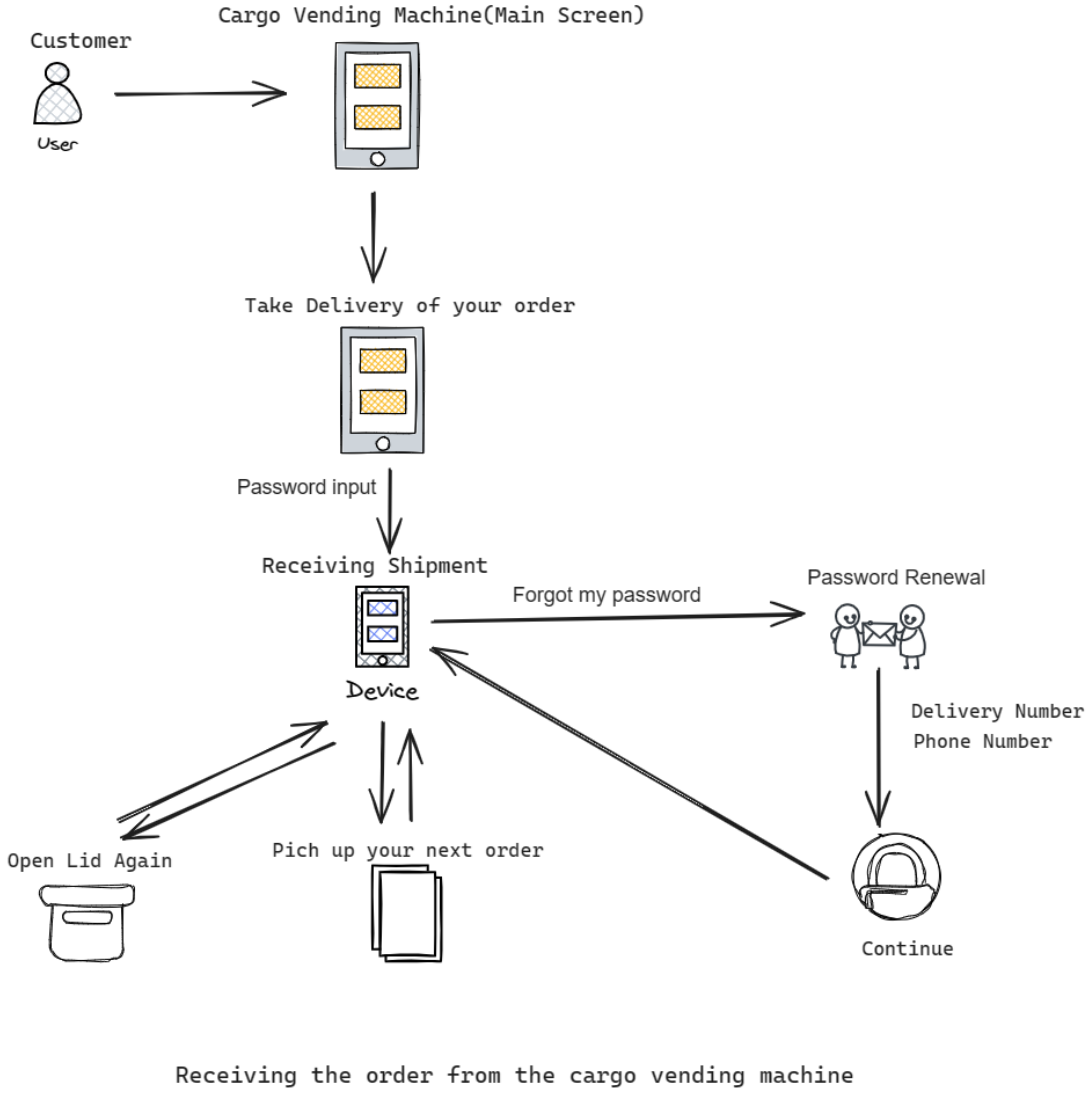

Figure 10: The process of customers picking up their orders from the cargo vending machine

When the customer presses the "*TAKE DELIVERY OF YOUR ORDER*" button on the main screen of the cargo vending machine shown in Figure 5, they will be directed to the password entry screen shown in Figure 11.

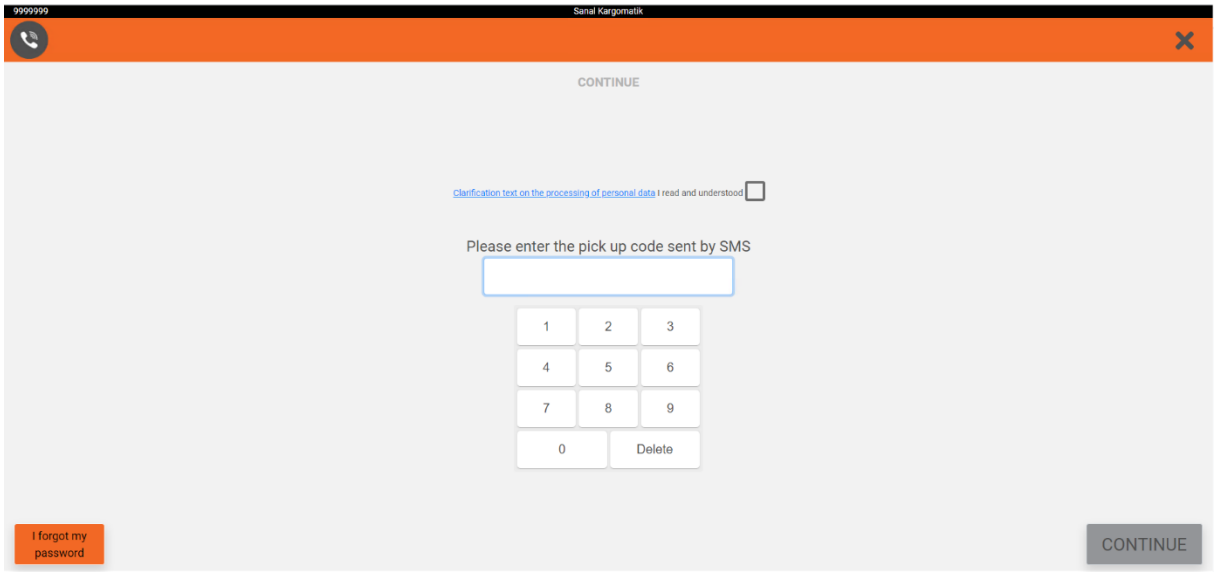

Figure 11: Password login screen

On this screen, the user can first read the Clarification Text on the Processing of Personal Data and then is expected to check the "I have read and understood" box. If the user does not check this box, the transactions cannot continue. After checking the relevant box, the user enters the password sent by the e-commerce company and passes to the "Order Receiving" screen shown in Figure 12.

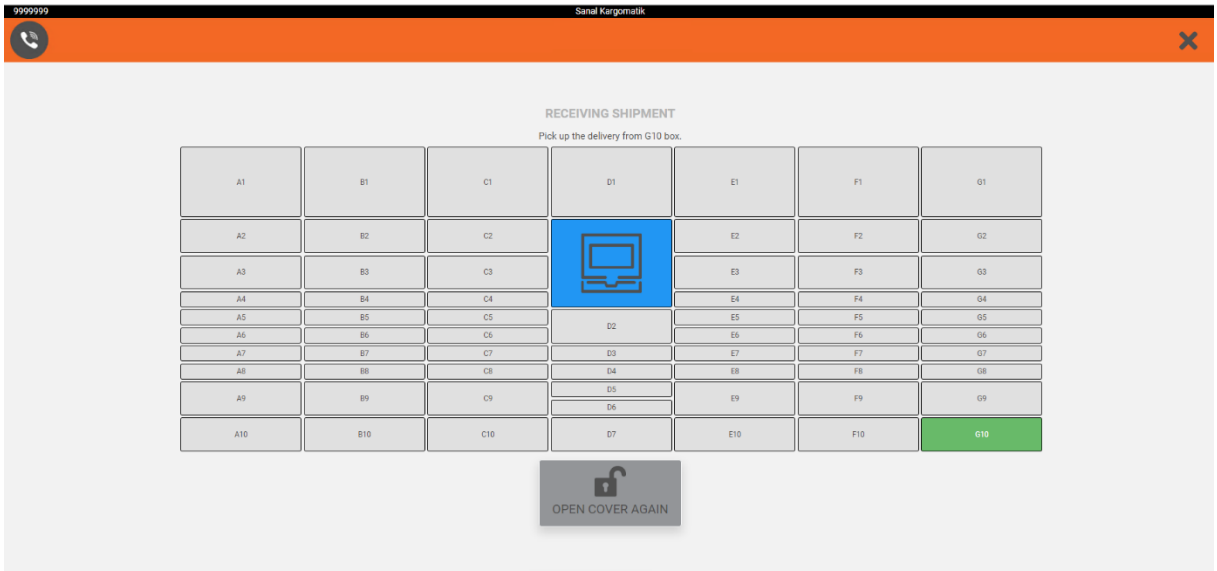

Figure 12: Order receiving screen

The hopper from which the customer will pick up his/her cargo is colored green as shown in Figure 12. In this case, the lid opens by itself and an audible warning is given as "*Please pick up your shipment*". In cases where the lid cannot be opened or jammed, the "*OPEN LID AGAIN*" button has been added. This button works as described in the title of placing the shipment in the cargo vending machine. After the customer takes the shipment from the hopper, an audible warning is given as "Please close the lid" and the transaction is considered successfully completed. In the case of multiple product pick-ups, all steps are executed in the same way, only after the product from the first hopper is picked up, the hopper with the other product is opened by clicking the "*PICK UP YOUR NEXT ORDER*" button as shown in Figure 13. In this way, the transaction is considered successful after all shipments have been received.

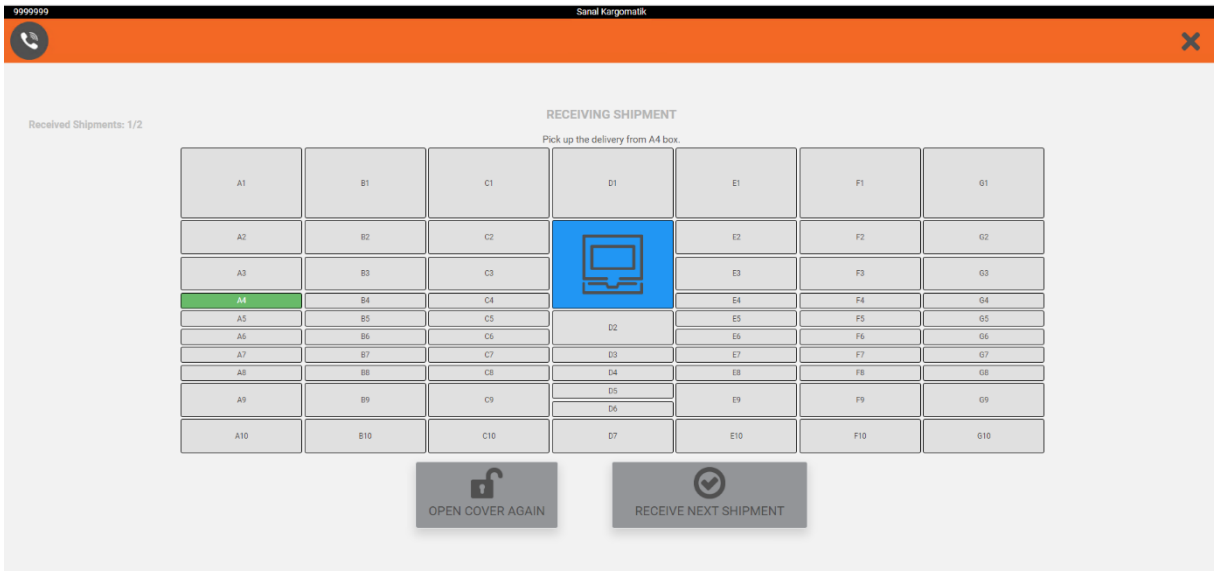

Figure 13: Multiple order receiving screen

In the screen shown in Figure 11, if the customer forgets his/her password, he/she can click on the "*FORGOT MY PASSWORD*" button. In this case, the customer is presented with the Password Renewal screen shown in Figure 14.

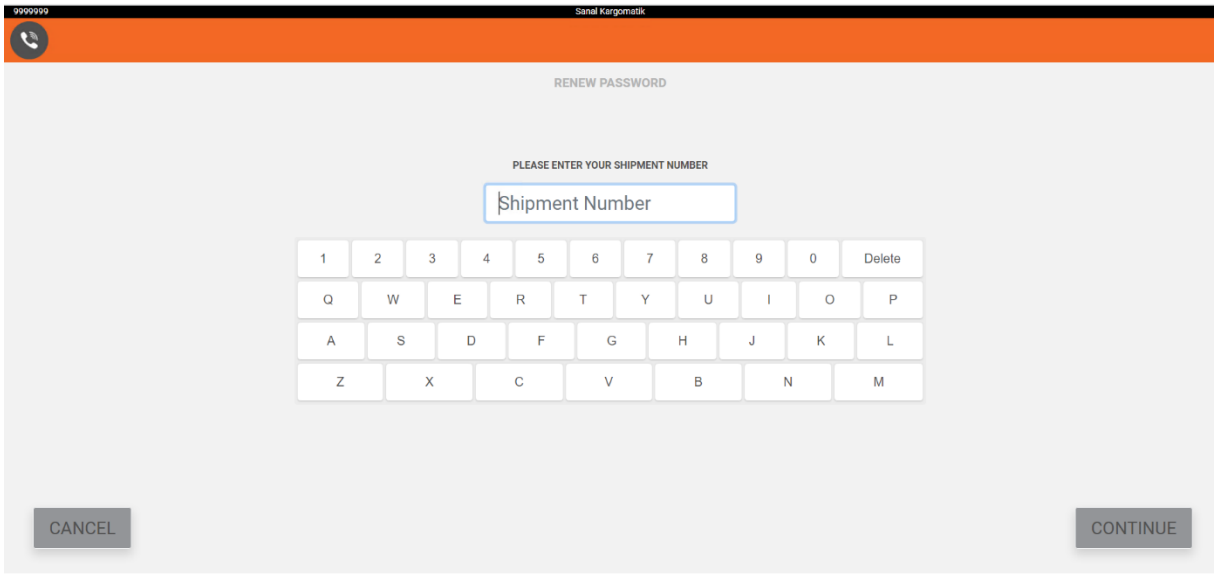

Figure 14: Password reset screen

On this screen, the customer first enters the Delivery Number and presses the continue button. The customer is presented with a screen where he/she enters the last two digits of the phone number shown in Figure 15. After pressing the Continue button, a new password is sent to the customer's cell phone.

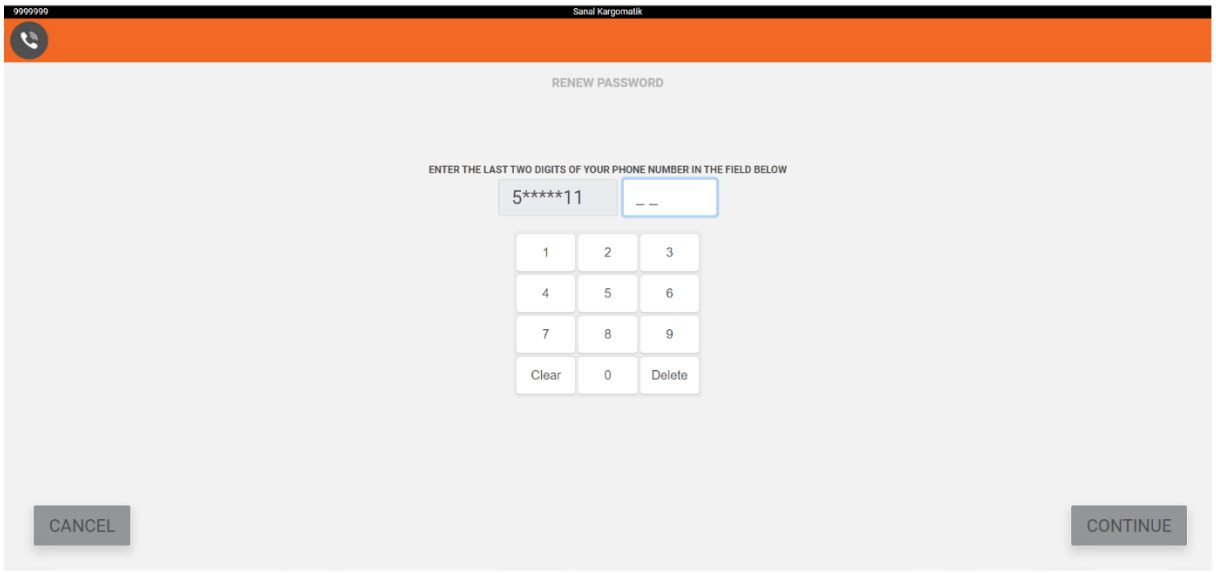

Figure 15: The screen where the last two digits of the phone number are entered

The phone icon in the upper left corner of the screen shows the call center contact number of the e-commerce company and the steps to be followed in the application.

2.4.3 Order refund process

The order refund process is initiated when the customer orders are placed in the cargo vending machines and are not received at the end of the periods determined by the e-commerce company. The refund processes are initiated by the e-commerce company. The e-commerce company transmits the refund process information of the relevant order to the server to which the relevant cargo vending machine is connected and is transferred to the cargo vending machine device via the service running on this server. When the cargo personnel comes to place the product in the relevant cargo vending machine, they enter the reconciliation number given to them. If there are products to be returned first, the screen shown in Figure 16 appears before the cargo personnel.

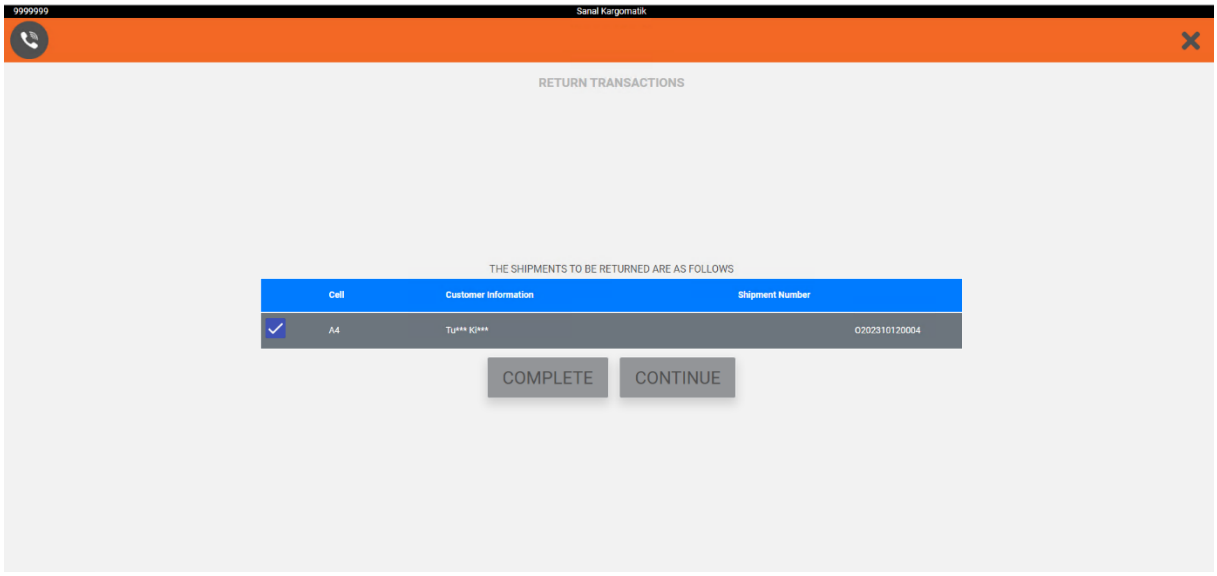

Figure 16: Cargo personnel refund processing screen

This screen contains information about the products to be returned. By pressing the "*CONTINUE*" button on this screen, the screen shown in Figure 17 appears before the cargo personnel.

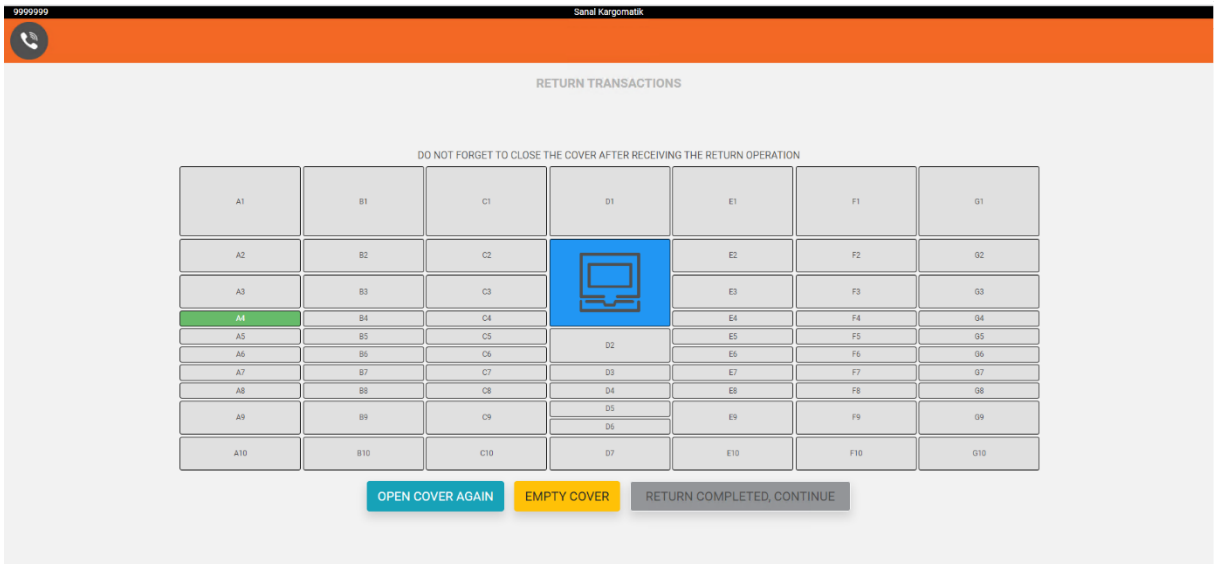

Figure 17: In refund processes, the screen for retrieving the order from the hopper

The product to be refunded is in the green colored hopper. The relevant hopper opens automatically. After the cargo personnel receive the refunded products, the process is continued by closing the hopper cover and clicking the "REFUNDE RECEIVED CONTINUE" button. On the next screen, a screen showing the summary of the refunded products is displayed. Cargo personnel complete their transactions by comparing the refunded products on this screen. After this process, the SMS password sent to the relevant customer is invalidated and no delivery is made to the customer. Then, if the cargo personnel will place a new order, they can continue to perform these operations.

Even if the refund process is initiated by the e-commerce company, the customer can pick up the order from the cargo vending machine during the period until the cargo personnel receive the order from the relevant cargo vending machine. If the customer picks up the order at the cargo vending machine during this period, the return process is automatically canceled.

2.4.4 Order cancellation process

Cancellation can be initiated at any stage after the creation of the order through the e-commerce company. Cancellation procedures are initiated for reasons such as loss of the customer order, damage to the order, fraud (online purchases made by copying, stealing credit card information in e-commerce or taking information from the cardholder with various fraud methods) and the customer canceling the order. The cancellation is made by the e-commerce company and then the refund process is initiated. After this process, the system prevents the delivery of the order to the customer. Cargo personnel are directed to follow the steps in the return process and the process is finalized by following the necessary steps.

2.4.5 Service operations

There is a "SERVICE OPERATIONS" button on the main screen shown in Figure 5. This menu has been created for cleaning staff to perform cleaning operations and technical service personnel to perform maintenance, repair and control operations. When this button is pressed, the Service Operations screen shown in Figure 18 appears.

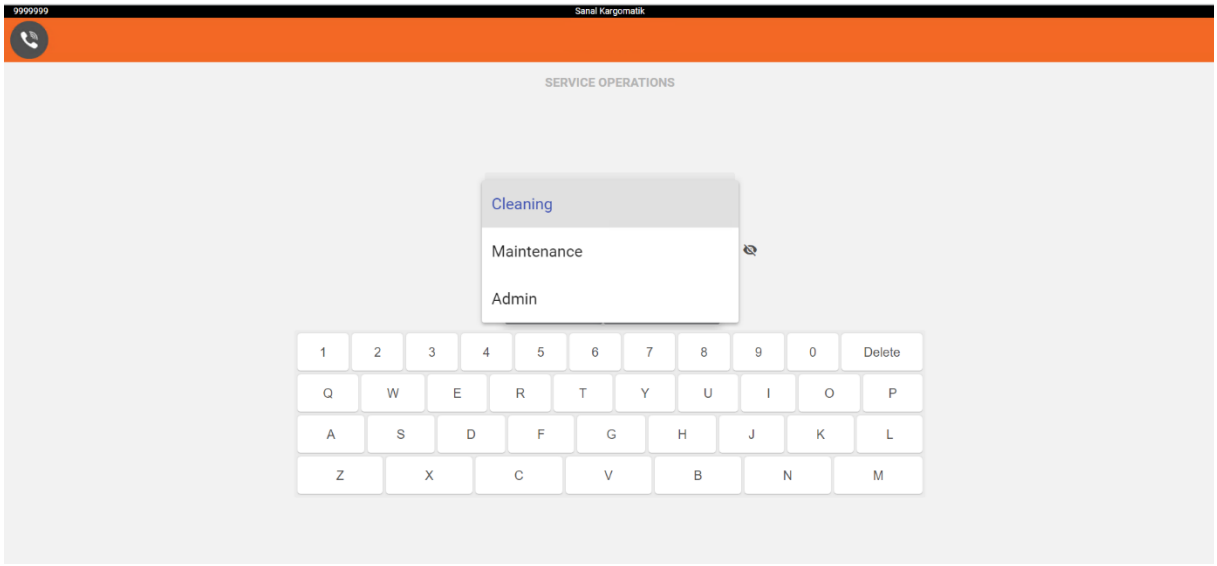

Figure 18: Service operations screen

There are three different types of users on this screen. Each group selects their own user name and after entering their passwords, they perform their transactions related to their own fields. The passwords of the individuals are generated automatically through the main system at certain intervals. These generated passwords are sent to both the device and the relevant persons via e-mail.

2.5 Indoor Cargo Vending System Software

Many information such as reconciliation number, customer password, cleaning and technical staff passwords, return information, cancellation information, etc. must be transferred to the cargo vending machines. It is necessary to know clearly that this information is fully transferred to the cargo vending machines and that the transfer is transferred correctly. If there are deficiencies in this information, it may cause problems such as the customer not receiving the product and technical personnel not being able to open the cargo vending machines. Services and software developed to prevent such problems are described under this heading.

2.5.1 Integration processes after order placement

In shipment placement processes, all information to be transmitted to the cargo vending machine is transmitted by e-commerce companies to a central server to which the cargo vending machines are connected. The reason for this is that e-commerce companies can transmit data directly to cargo vending machines, but since they cannot guarantee that the data is transmitted exactly, all data is transmitted to a central server. On this server, services have been prepared that guarantee that the data that needs to be transmitted is transmitted completely and accurately. While the passwords required for the customer to receive their orders are transmitted to the customers as SMS by the e-commerce company, the same password is transmitted to the cargo vending machine via the central server. When the customer enters the password on the cargo vending machine, verification can be done. All these steps are explained on the architecture shown in Figure 19.

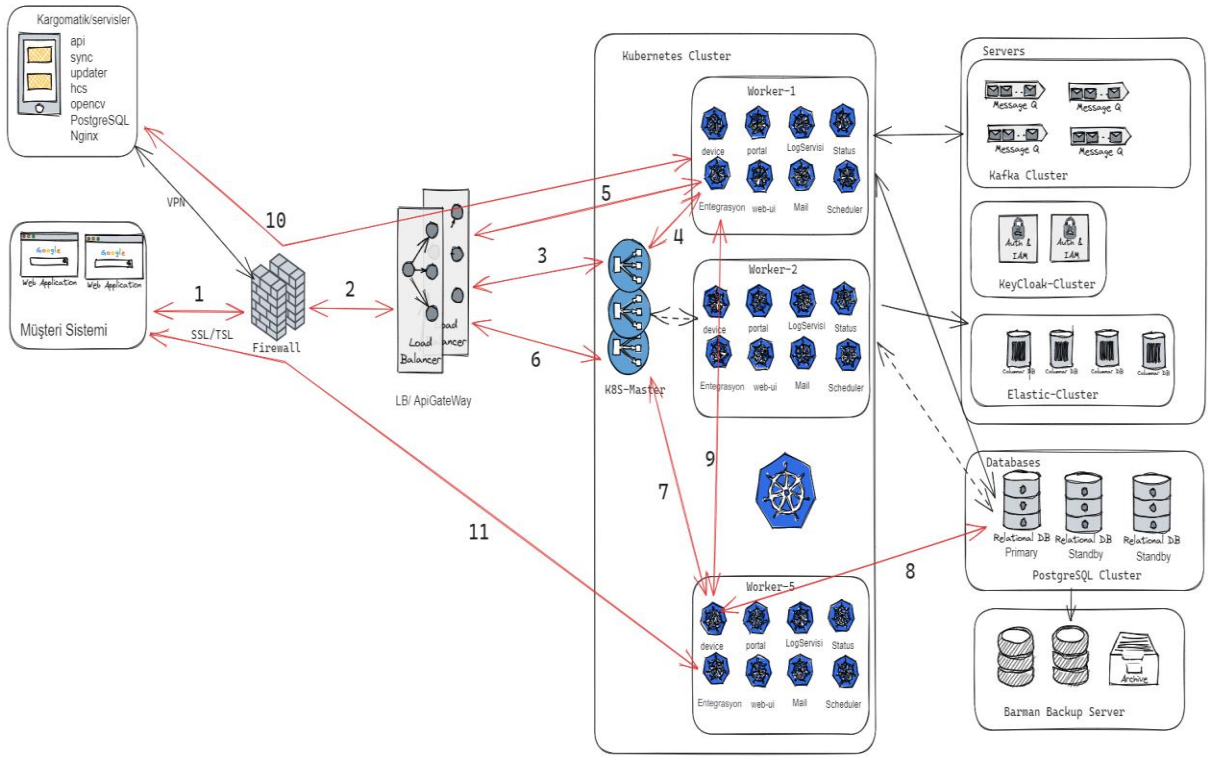

Figure 19: The process of transmitting the package information from the e-sales application to the cargo vending machine

The operations in the numbered stages in Figure 19 are explained below.

Process-1: A request is sent to the integration service by the e-commerce company. Incoming requests are met by the firewall and it is confirmed whether they come from an authorized system.

Process-2: If a successful pass through the firewall is provided, the request is directed to the api gateway. After Authentication check on the api gateway, the request is directed to Load balancer.

Process-3: Load balancer sends the request to the active Kubernetes master.

Process-4: Kubernetes master routes the request to the active Kubernetes worker. Kubernetes worker redirects the request to the service belonging to the domain.

Process-5: Integration service sends a request to the device domain via load balancer to learn the ip address of the cargo vending device by looking at the content of the incoming request.

Process-6: Load balancer sends the request to the active Kubernetes master.

Process-7: Kubernetes master forwards the request to the active Kubernetes worker. Kubernetes worker redirects the request to the service belonging to the domain.

Process-8: Device service reads the ip information via Postgresql and returns a response.

Process-9: Integration service receives the response on the same channel.

Process-10: Integration service sends a package information request to the cargo management device. Transaction confirmation information is received via the cargo vending device.

Process-11: Transaction confirmation information is sent to e-commerce service.

2.5.2 Used applications

Services and applications used in cargo vending systems and their intended use are described below.

Cargo vending-api: It is the backend software on cargo vending machines. The application processes request from the interface and forwards them to the database and the relevant server software. It directs requests such as opening/closing hopper lids to the relevant hardware service. Manages requests by connecting with the hopper control unit. Responds to requests from the server. Processes sensor data in the hoppers and processes status information such as open/closed, full/empty.

Cargo vending -UI: Provides an application interface for users. Nginx is used as web server. It opens automatically in kiox mode on chromium every time the PC starts.

Cargo vending -hcs: Provides access of API software to cargo vending hopper and lids. Controls all hardware.

Cargo vending -sync: It is the service that transfers the "log/transaction" information generated by the API to the server side at short intervals. These logs are written to the kafka queue via the log service on the server side.

Cargo vending -opencv: It contains hundreds of functions that support capturing, analyzing and manipulating visual information connected to a computer by webcams, video files or other types of devices thanks to the OpenCV library.

Cargo vending -updater: Starts the update process by following the update tasks created through the portal. By getting the name and version information of the service to be updated, it pulls the update on the cargo automaton via Object Store and restarts the service.

Cargo vending machine sleep services: It enables cargo vending machines to be put into sleep mode (suspend) at night and to be woken up automatically at the specified time in the morning. During sleep, the customer can wake up by touching the screen if necessary. After the customer's transaction is completed, it goes back to automatic sleep after 3 minutes of passive waiting.

PostgreSQL: Due to the offline working structure of the cargo vending, a database is run on them. Two separate databases are defined for API service and updater services.

Nginx: Serves as a web server.

2.5.3 Specially developed applications

The applications and services described here have been developed to control and provide solutions to a number of major problems.

Updater application: It is an application that works in a structure similar to Ansible but developed as custom. Work tasks are defined to cargo vending machines via the Portal. Cargo vending machines regularly check their work tasks and make updates. The results are also tracked on the relevant portal page.

Uploader application: It is a custom developed software that allows uploading software packages or script files to be updated to the Object store.

Remote PDU control: All devices in the cargo vending machine and connected to the PDU can be turned off and on with this application. The computers on the cargo vending machine, which we cannot connect via remote ssh, can be turned off / on via PDU (Smart Socket) when necessary.

Portal warning monitoring application: It is the portal screen where the status of all cargo vending machines are monitored via the portal and detailed failure logs are viewed.

Energy saving software: In order to save energy, cargo vending machines were suspended at a specified time interval. Customers can wake up by touching the screen when necessary. In the event that the full operation of the wake-up scripts coincides with the moment of package delivery or package placement, due to the negative impact on the operation of the cabinet, it is ensured that extra scripts are researched and written to delay the sleep process by determining whether the cabinet is being used at that moment.

3 Conclusions

With the development of e-commerce, the importance of the logistics sector also increases. In cases where e-commerce companies deliver their orders to their customers and receive their returns, the orders are sent back to the e-commerce companies in cases such as the customer not being at the address or changing the address, which increases the shipping costs. Similarly, it creates problems such as customers not being able to receive their orders from cargo companies.

Indoor cargo vending systems have been developed to overcome similar problems. These systems are installed at certain points and allow cargo personnel and customers to go and pick up their orders independently of time. In the developed system, a central server system has been prepared to which all cargo vending machines are connected. The system allows for expansion depending on the number of cargo vending machines. Thanks to the offline operation feature and suspend modes of cargo vending machines, it keeps customer satisfaction at the highest level with both energy efficiency and service uninterruptedness. In addition, all cargo vending machines are constantly monitored through a central portal and all relevant parties are informed instantly in case of malfunctions. Cargo vending machine software can be updated from the central point. In order to ensure that the updates are not left unfinished, the current files are first transferred to the cargo vending machine. When the information that all files are transferred completely, the update works on the vending machine and a problem-free process is realized. In addition, the information received from e-commerce companies is transferred to the relevant cargo vending machines smoothly with the system software developed. If the transfer is interrupted, the file on the cargo vending machine side is deleted and the relevant files are started to be transferred again after the connection is re-established. Thus, it is guaranteed that the data is transferred completely and accurately.

4 Declarations

4.1 Study Limitations

The study's findings are based on the implementation and testing of a prototype in a controlled environment. The scalability and generalizability of the system to different settings and scales, such as urban and rural areas, diverse e-commerce businesses, and varying customer demographics, need further exploration.

4.2 Acknowledgements

The authors and the project team would like to thank Canovate Electronics Industry and Trade Inc. and R&D Center for their contributions.

4.3 Funding source

This scientific research project work was supported by Canovate Electronics Industry and Trade Inc. R&D Center with the project number of M22-144 and titled "Indoor Smart Autonomous Cargo Delivery Cabinet Design".

4.4 Competing Interests

There is no conflict of interest in this study.

4.5 Authors' Contributions

Define the contribution of each researcher named in the paper to the paper.

Corresponding Author Süleyman UZUN: Contribution to the article. Organizing and reporting the data, taking responsibility for the explanation and presentation of the results, taking responsibility for the literature review during the research, taking responsibility for the creation of the entire manuscript or the main part, reworking not only in terms of spelling and grammar but also intellectual content or other contributions.

2. Author's Mustafa Zahid YILDIZ: Contribution to the article. Organizing and reporting the data, taking responsibility for the explanation and presentation of the results, taking responsibility for the literature review during the research, taking responsibility for the creation of the entire manuscript or the main part, reworking not only in terms of spelling and grammar but also intellectual content or other contributions.

3. Author's Tolga SUGETİREN: Contribution to the article. Developing ideas or hypotheses for the research and/or article, planning the materials and methods to reach the results, taking responsibility for the experiments, organizing and reporting the data, taking responsibility for the explanation and presentation of the results.

4. Author's İsmail YÜKSEL: Contribution to the article. Developing ideas or hypotheses for the research and/or article, planning the materials and methods to reach the results, taking responsibility for the experiments, organizing and reporting the data, taking responsibility for the explanation and presentation of the results.

5 Human and Animal Related Study

The work does not involve the use of human/animal subjects.

5.1 Ethical Approval

No ethical approval is required for this project.

5.2 Informed Consent

All authors consent to the publication of the study.

References

[1] Karabulut OA, Seyret MK, Avcı MG. Kargo Otomatı Yer Seçimi Problemi için Bir Matematiksel Model. Endüstri Mühendisliği 2023;34:26–41. https://doi.org/10.46465/endustrimuhendisligi.1130216.

[2] Rohmer S, Gendron B. A guide to parcel lockers in last mile distribution: Highlighting challenges and opportunities from an OR perspective 2020.

[3] Tang YM, Chau KY, Xu D, Liu X. Consumer perceptions to support IoT based smart parcel locker logistics in China. J Retail Consum Serv 2021;62:102659. https://doi.org/10.1016/J.JRETCONSER.2021.102659.

[4] Toraman Y. Planlı Davranış Teorisi Çerçevesinde Kargo Otomat (Kargomat) Sisteminin İncelenmesi: Elektronik Son Adım Teslimat Süreçlerinin E-Ticaret Platformları Özelinde Araştırılması. J Transp Logist 2022;7:303–20. https://doi.org/10.26650/jtl.2022.1006593.

[5] YAMAN O, BAYĞIN M. UHF-RFID Based Smart Cargo Management and Real Time Tracking Approach. J Intell Syst Theory Appl 2020:38–45. https://doi.org/10.38016/jista.762685.

[6] Kim K, Kim J-H. A Development of an Low Cost Smart Parcel Service System with Enhanced Security. J Converg Inf Technol 2018;8:193–9. https://doi.org/10.22156/CS4SMB.2018.8.6.193.

[7] Çakır A, Güngör O. Rfid ile Kargo Yönetimi. Afyon Kocatepe Üniversitesi Fen Ve Mühendislik Bilim Derg 2011;10:83–9.

[8] Atlassian. Jira Work Management | A Friendly and Powerful Way to Work. Atlassian 2023. https://www.atlassian.com/software/jira/work-management (accessed September 30, 2023).

[9] Filion L, Daviot N, Le Bel JP, Gagnon M. Using Atlassian tools for efficient requirements management: An industrial case study. 11th Annu IEEE Int Syst Conf SysCon 2017 - Proc 2017. https://doi.org/10.1109/SYSCON.2017.7934769.

[10] Chakraborty S, Aithal PS. A Practical Approach to GIT Using Bitbucket, GitHub and SourceTree. Int J Appl Eng Manag Lett 2022;6:254–63. https://doi.org/10.47992/IJAEML.2581.7000.0156.

[11] Demiral Y, Carkaci N, Cekmez U. DevOps architecture in the cloud. 27th Signal Process Commun Appl Conf SIU 2019 2019. https://doi.org/10.1109/SIU.2019.8806433.

[12] HANAYLI M. GİT: Versiyon Kontrol Sistemi 2022.

[13] Smart J. Jenkins: The Definitive Guide: Continuous Integration for the Masses 2011.

[14] Rensin DK. Kubernetes 2015.