

Reversing Negative Trends in Spare Parts Control

Fernando W.A.M.A.R.¹, Jinadasa U.G.O.C², Amarasinghe B.P.A³, Mandalawatta M.T⁴, D.I de Silva⁵ and Dulanji Cooray⁶

¹Department of Computer Science and Software Engineering, Sri Lanka Institute of Information Technology, SRI LANKA

²Department of Computer Science and Software Engineering, Sri Lanka Institute of Information Technology, SRI LANKA

³Department of Computer Science and Software Engineering, Sri Lanka Institute of Information Technology, SRI LANKA

⁴Department of Computer Science and Software Engineering, Sri Lanka Institute of Information Technology, SRI LANKA

⁵Department of Computer Science and Software Engineering, Sri Lanka Institute of Information Technology, SRI LANKA

⁶Department of Computer Science and Software Engineering, Sri Lanka Institute of Information Technology, SRI LANKA

²Corresponding Author: preewije@gmail.com

ABSTRACT

The consumer SPARE BROTHERS' a Colombo, Sri Lanka company, commissioned the implementation of the online e-commerce management system. The users currently manage its employees' significant amounts of data manually using a manual approach. The suggested solution enables consumers to store and manage data in own database, greatly improving the efficiency of their data handling methods. The suggest system is designed to handle all the steps involved in managing a spare parts business, including consumer, store, cart, payment, delivery, order, and rider management. Firstly, if a user is already a consumer, user can connect to the system with users' credentials and be redirected to user's responsible side. If not, user can sign up for the system. Here, it includes with this system set up as a distinct site for the user and the administrator. The system's spare parts can be added by the admin, and those parts will then appear on the user side. Hence that consumer can purchase products based on user's interests. Users can manage cart items here, and if user's wish to proceed to check out, users can enter own payment and delivery details. After the order is complete, the administrator can designate a rider to transport those things to the appropriate location. Lastly, the admin can edit, delete, and generate certain essential reports to manage all the user, product, order, delivery, and rider details. Consequently, the administrator can add notifications, search relevant categories, and allow users to add wish lists. The solution decreases data manipulation errors in the present process, leading to a significant improvement in the utilization of human resources. It enhances consumers' interactions with their customers, increasing revenue for the company. The process is simple to use, dependable, and effective, and its statistics meet the needs of consumers.

Keywords-- Spare Parts, E-Commerce, Online, Management System, Ordering

facilities and to quickly order and pay for such facilities without having experienced any difficulty. Online e-commerce spare part application is one of the amazing features of the web. Automated ordering spare parts is not only fast, simple, and convenient, but also impressively affordable. Nowadays, most businesses have websites, providing consumers to visit them and review the features and facilities that each one has to offer. Compared to the current manual approach for processing consumer-used spare parts, the suggested automated online e-commerce method is intended to be more useful, organized, efficient, quick, and precise. As the perspective goes, every spare parts store should use this method. The e-commerce platform was built using Express JS, Node JS on the backend, Rest API, React JS on the frontend, and the database has been Mongo DB. Consumers can search for and order the appropriate spare parts in consistent with users demands, using this system, which also enables Admins to create, update, and delete users such as customer, admin, and rider, spare parts, items, payments, and deliveries. If a user does not already have a profile, users must create one before utilizing the system. After successfully registering, the consumer will have access to the task in the system. Consumers that successfully log in are redirected to the main page, where user can view the available different products. Products that pick will be put in the shopping cart. And pay for the items which are ordered. The administrator can control the entire user-side operations in this section. The administrator will be responsible for entering pricing adjustments for products and adding them to the system. The major function of this page is for the administrator to give the riders tasks. Additionally, the rider may view the order on the admin side, ensuring that the delivery will be made correctly and to the correct location. The administrator can obtain both hard copies and soft copies of the entire system and have a comprehensive understanding of how the company is doing right now.

I. INTRODUCTION

The purpose of this project is to support a spare parts store in maintaining the basic data about parts and accessories. The major goal of this project is to make it simple for consumers to browse the spare parts shop and the

II. RELATED WORK

Additionally, the problem of product failure with spare components frequently results of their being destroyed at a very low value. Overall, managing spare part inventories is critical to ensuring that the required product is available at the lowest economic and environmental cost. However, managing spare components is not without its challenges[2]. So those failure might be reduced through this implemented system.

There should be a contrast between maintaining a supply of replacement parts on board a ship and maintaining a supply of spare parts on land, in warehouse facilities. To save additional delivery fees, stock outs, and schedule delays, it is ideal to have all necessary stock on board the vessel[3]. In this integrated system only implements for local business and by delivery planed for throughout the country.

Some of these factors, such repair times and spare part requirements, can be assessed for current systems during operation to reduce the cost of the product life cycle. The evaluation of field data assists the designer and engineer in modifying the design and/or product support strategy to increase system dependability and determine the necessary spare parts[4]. In this system for the efficient of

the users it implements some calculation methods in the add to cart function and it will be engaged with the payable total amount.

SPM has a lot of promise now for analyzing the internet revolution, foreign suppliers, quick communications, and tracking shipping orders. Most businesses have not fully embraced the advantages of the newly available SPM development avenues, albeit may have become aware of some of them. Every business has its own rules, regulations, and goals for enhancing the management system[5]. By using an e-commerce app for this industry is very useful in this situation. It will gain the advantage not only for the shop owners but also for the consumer will not be waste their time.

III. METHODOLOGY

Technologies

The Spare Parts Management platform was driven by Express JS, Node.js on the application server, Rest API and ReactJS on the client end, and MongoDB.

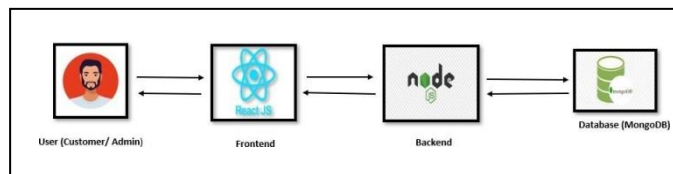


Figure 1: Technology Flow

React JS

An open-source and cost-free JavaScript library is React.js, sometimes referred to as React[5]. Combining small code fragments (components) to create full websites is the greatest method for creating UI. Facebook produced it, and it is now maintained by the open platform.

React.js is built on JSX, a JavaScript/XML hybrid. The elements are built with JSX, and it rendered on the website with JavaScript. React is gradually rising to become one of the most well-liked and demanded Website builders, even though it could be difficult for a novice developer to understand. Although react is defined as a JavaScript library, the other solutions examined today are considered frameworks. A methodology is a graphical, whereas a catalog is a tool that can be used in any project by developers.

Express JS, Node JS

A set of tools are available for the creation of both web-based and mobile applications through a web browser technique called Express. It could be used to make a single-page, multi-page, or hybrid web application. Node JS's server and route management tiers work together. Building APIs and web apps is now easier thanks to Express. It

provides top-notch online and mobile applications while effectively halving the amount of coding required. Express has the added benefit of being developed with JavaScript, which even those without programming experience can easily learn. Express helps many new developers to break into the software development field.

The following characteristics are used to justify the creation of an express methodology for Node JS: it is quick, affordable, simple to learn, and sequential.

Rest API

REST is an architectural style that outlines restrictions, like the consistent interface, that, when applied to a web service, result in desired properties, including performance, scalability, and modifiability, that enable services to operate as efficiently as possible on the Web. Because they may be accessed using To act on the resources, a set of simple, clearly defined operations can be used. If a RESTful API can consider the following things:

- Client-Server is a constraint that indicates that the consumer and server should be stored and should be permitted to evolve independently.
- Statelessness is a condition. Due to the statelessness of REST APIs, calls can be made separately and yet contain

all the information necessary for the call to be successful.

- Every response must specify if it can be cached and for how long. The client will use its cache to deliver data for subsequent requests, negating the need to contact the server once more.
- Decoupling the client from the server necessitates a consistent interface that allows for individual application evolution without connecting, the services, or models and actions, to the API layer itself.
- An application architecture needs multiple levels. Between the client and the end server, there are numerous intermediary servers, and each layer only has access to the one it is now on. Intermediary servers can improve system availability by enabling load balancing and providing shared caches.

Mongo DB

Large volumes of data can be stored in the document-oriented NoSQL database MongoDB. As opposed to conventional relational databases, which employ tables and rows, MongoDB uses collections and documents. Key-value pairs, the basic building block of MongoDB data, make up documents. Since collections include groups of documents and functions, they are the relational database counterpart of tables.

The essays will be submitted in Binary JSON format, a type of JavaScript Object Notation (BSON). BSON has the advantage of being able to support additional data types. The fields in these documents are organized similarly to the columns through relational databases. The user guide for MongoDB states that contained values may include other documents, clusters, and clusters of documents, as well as any category variable. The identification of documents will be done using a primary key.

MongoDB, unlike other NoSQL databases, does not require specified schemas. It keeps all forms of data. Users can add as many fields as they like to a paper, but MongoDB databases can grow more easily than relational databases can. The advantage of documents is that they map to many programming languages' native data types. Additionally, fewer database joins are needed when documents are embedded, which can reduce costs.

IV. PROPOSED SYSTEM

To implement the system, the browser server model will be used. The system primarily manages User, Cart, Product, Payment, Delivery, Rider, and Admin.

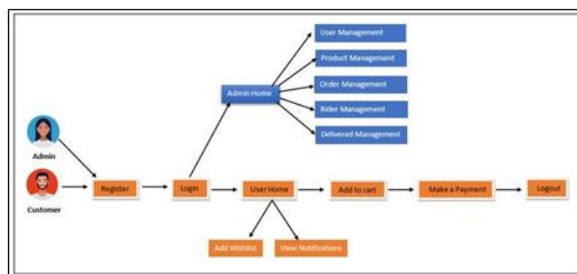


Figure 2: System Overview

Home

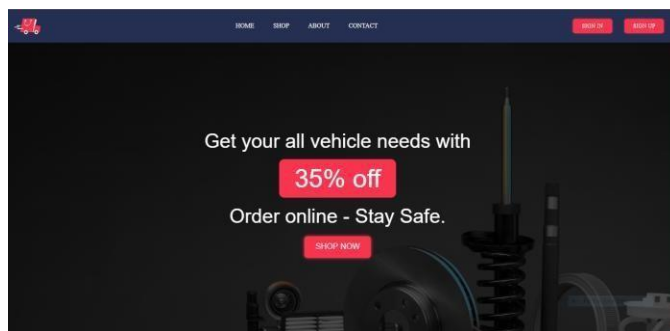


Figure 3: Home user interface

User Management

This function mainly focuses on user registration, login, profile update and add wish lists. Admin can perform user search, deletion, and report generation operations.

Figure 7 is showing the signup interface, Figure 8 depicts the login interface, Figure 9 depicts the profile interface, Figure 10 depicts the view all users interface, and Figure 11 depicts the add Wishlist interface.

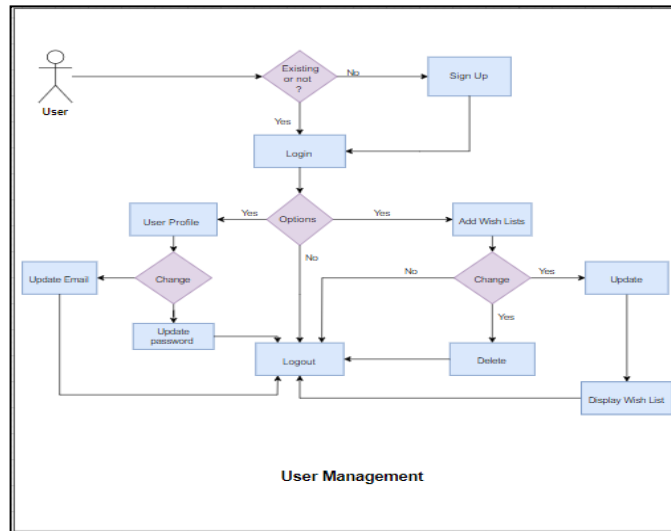


Figure 4: User Management flow

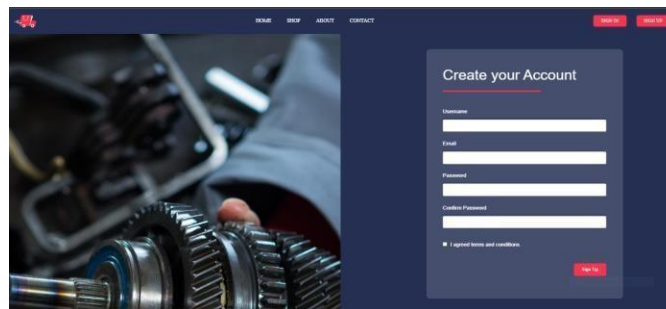


Figure 5: Sign Up user interface

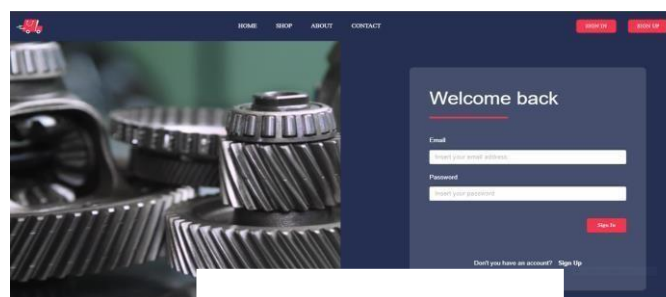


Figure 6: Login user interface



Figure 7: User Profile user interface

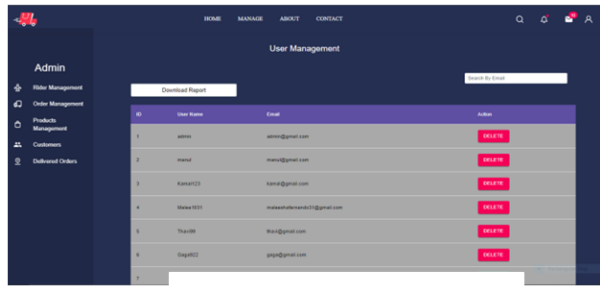


Figure 8: Admin User Management user interface

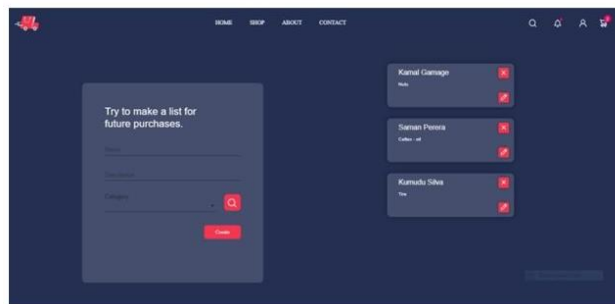


Figure 9: Wishlist user interface

Cart Management

This function mainly focuses on view products in category wise and add to cart. Admin can perform the update, delete, search and report generate operations of

products. Figure12 indicates the category products UI and Figure13 presenting the add to cart interface and Figure14 is showing the add product to the system in admin side interface.

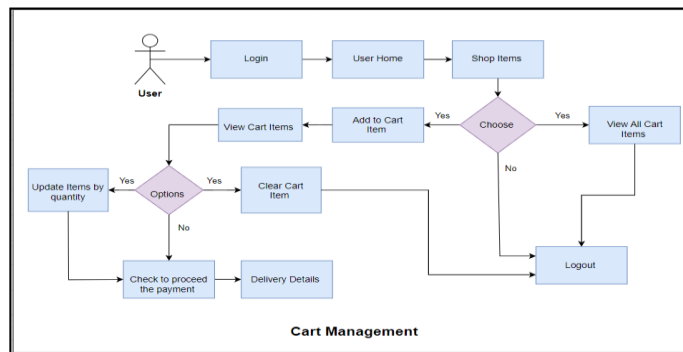


Figure 10: Cart Management

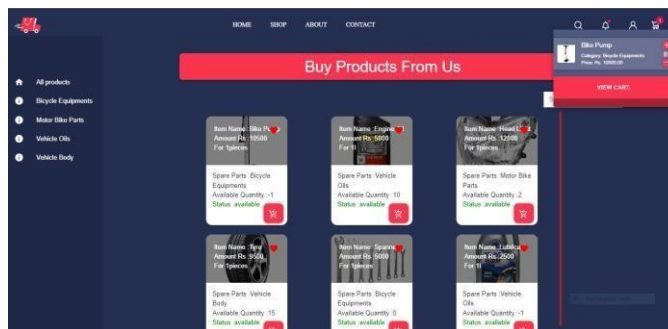


Figure 11: View Product user interface

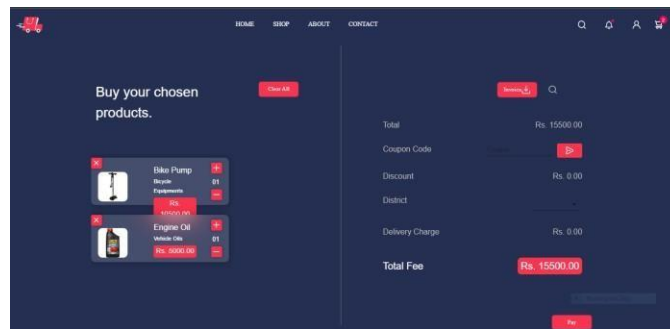


Figure 12: Add to Cart user interface

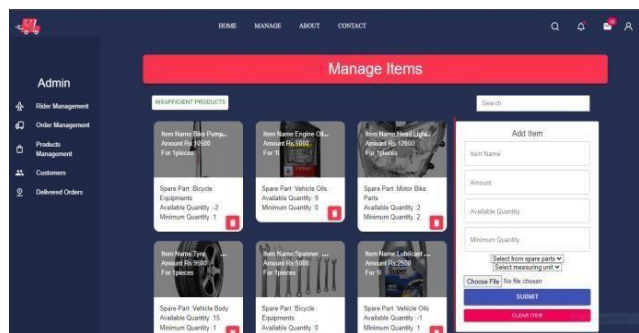


Figure 13: Admin Cart Management user interface

Payment Management

This function primarily focuses on adding delivery details and providing payment transaction details.

Figure 15 depicts the add delivery details interface, while Figure 16 depicts the payment interface.

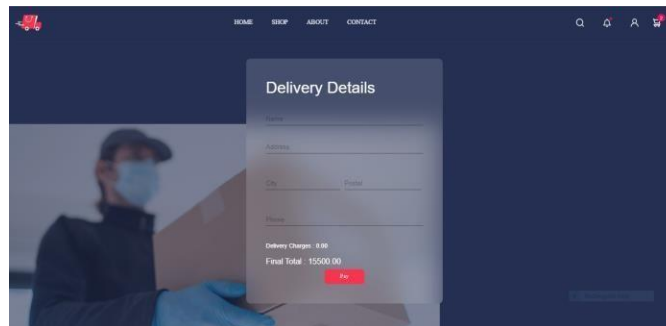


Figure 14: Add Delivery user interface

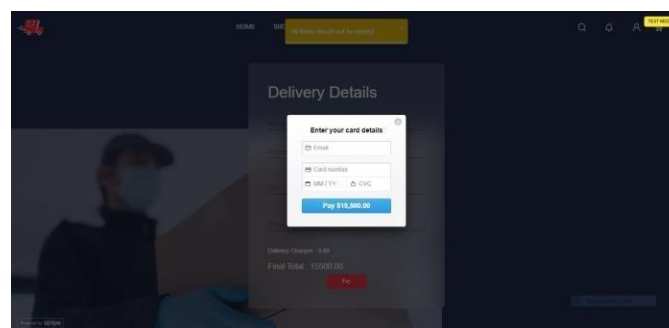


Figure 15: Add Transaction user interface

Order Management

This function primarily focuses on adding orders to riders, and the admin side can set, view, delete, search,

and generate reports. Figure 17 depicts the user interface for my orders, while Figure 18 depicts the admin interface for all orders.

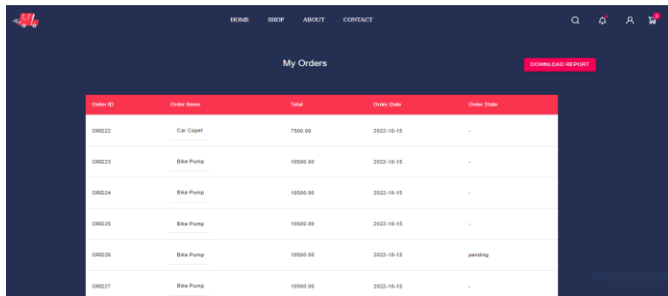


Figure 16: View My Orders user interface

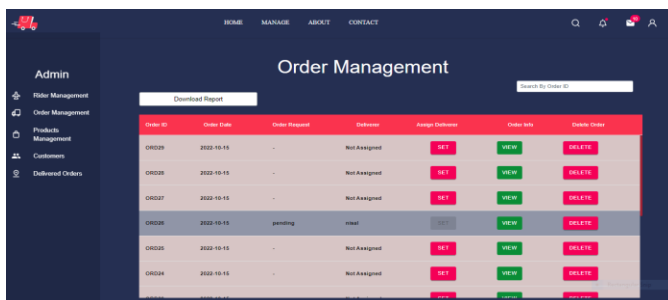


Figure 17: Admin Order Management user interface

Rider Management

This function mainly focuses on add riders,

update, and delete in admin side. Figure 19 is showing all rides interface in admin side.

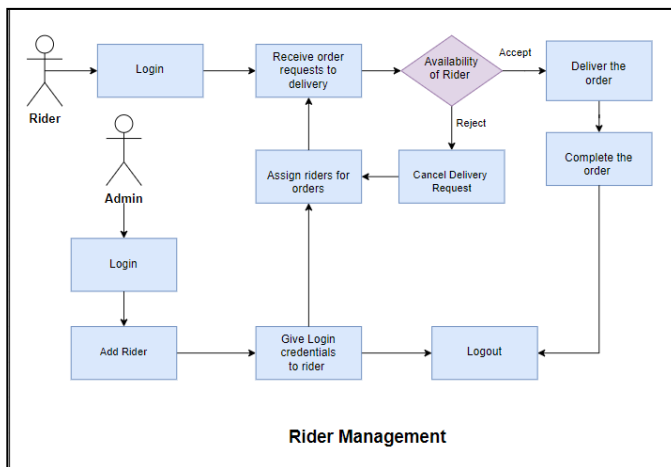


Figure 18: Rider Management flow

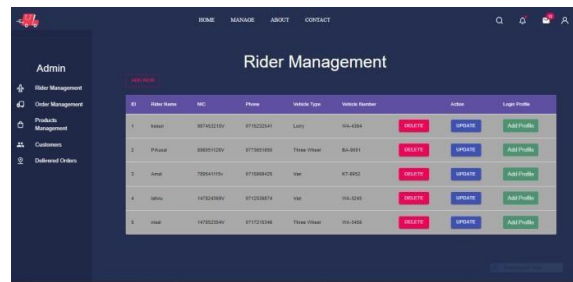


Figure 19: Admin Rider Management user interface

Delivery Management

This function mainly focuses on view delivered

orders, search, and generate annual report. Figure20 is showing all rides interface in admin side.

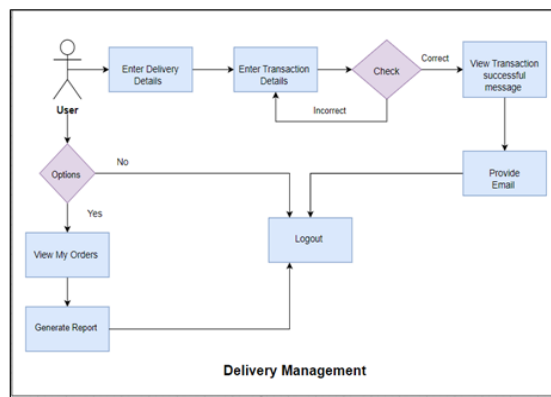


Figure 20: Delivery Management

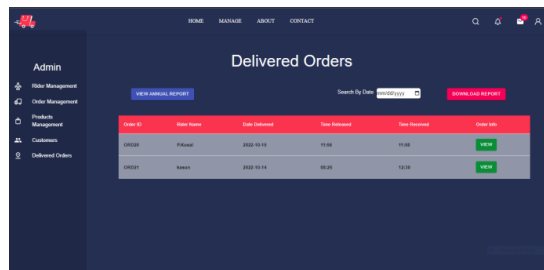


Figure 21: Admin Delivered Order Management user interface

Additional Pages

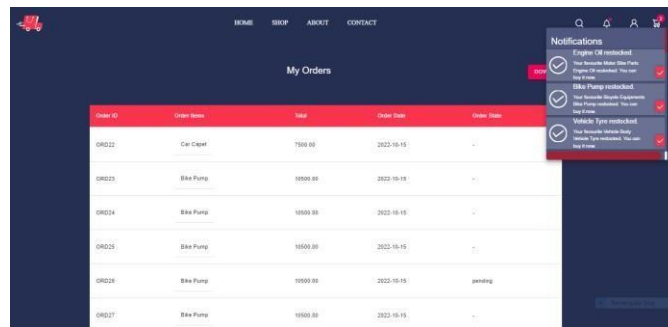


Figure 22: Additional Page user interface

V. DISCUSSION

This spare part system was created to eliminate the major issues with spare parts collection and processing. The COVID-19 pandemic situation with the current economic crisis may necessitate high transport costs. Consumer purchasing power slightly decreased because of rising fuel prices. Therefore, consumers using online platforms need not worry about those issues, because consumers are able to purchase their desired items on the online marketplace. It is an effective method for overcoming these situations and it processes extremely quickly. Since this approach enables consumers to feel free to deliver their essential accessories together with a wide range of spare items to the doorstep, and do not worry about the details. By utilizing those facilities, the business can expand its consumer base significantly, which will have an impact on its rivals as well. Payments on online platform are well secure, so do not be concerned. However, since delivery times are shorter, consumers can supply their goods much more rapidly. As the frontend and backend frameworks for the system design, React and Express, respectively. The mongo DB cloud database stores all the data in collections. For the implemented solution was developed using several libraries, including axios, React-from-validation, redux, mongoose, node mailer, message bird, stripe, and express. The designed interfaces are very user friendly and utilize icons to make them easy to comprehend. The interface was styled using Bootstrap and external CSS, while the icons came from the Front Awesome and Material UI.

VI. CONCLUSION

Sri Lanka is an e-commerce platform is now widely used for both purchasing and selling. By the way, it is a speedy way to buy and sell products for the busy life. Due to the COVID-19 pandemic, we encountered numerous challenges over the past two years, including numerous difficulties with the physical sale of goods. And this position was further complicated by the current economic crisis. Most businesses have switched to e-commerce systems, which are highly common in other industrialized nations, to deal with both downward issues.

The launch of this new platform benefits the owners greatly in terms of client base, and their retailer requirements also gradually rise. Modern technologies and features are not included in the systems that are currently in place, since include things are so outdated. Most of the issues now in existence are resolved using this system.

The user interface is incredibly straightforward, and icons are employed to make it simple for any user to understand. The interfaces were developed with attractive colors and pictures. As the project evolved, the primary objective was to reduce the number of employees and make them easier to manage. We can also provide a wide range of products in accordance with various categories, improving transaction security on the web platform. To increase productivity and customer satisfaction with this system, high technology was applied.

REFERENCES

- [1] A. S. E. W. M. M. C. v. d. H. R. D. E. Topana. (2019). A review of operational spare parts service logistics in service control. *Netherlands*.
- [2] S. & H. K. & Y. Y. Zhang. (2021). Spare parts inventory management: A literature review. *Sustainability*, 13, 2460. DOI: 10.3390/su13052460.
- [3] S. T. Mouschoutzi. (2020). *A comprehensive literature review on spare parts logistics management in the maritime*.
- [4] <https://www.diva-portal.org/smash/get/diva2:990793/FULLTEXT01.pdf>.
- [5] <https://www.diva-portal.org/smash/get/diva2:850914/FULLTEXT01.pdf>.
- [6] <https://reactjs.org/docs/getting-started.html>.
- [7] <https://www.mongodb.com/docs/>.
- [8] <https://docs.npmjs.com/>.
- [9] <https://stripe.com/docs/api>.
- [10] L. Vukić. (2021). *Modelling the optimal delivery of spare parts to vessels: Comparison of three different scenarios*.
- [11] <https://mui.com/>.