ISSN 2278-3091

Volume 9, No.3, May - June 2020

International Journal of Advanced Trends in Computer Science and Engineering

Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse144932020.pdf

https://doi.org/10.30534/ijatcse/2020/144932020

Constructo App: A Multi-platform Android and Web Application with Single Page Architecture for Ordering of Building Materials

Lloyd Kristoper Lim[1], Leopoldo Abing Jr.[2], Pitz Jerald Rabe[3], Jan Carlo T. Arroyo[4], Allemar Jhone P. Delima[5]

[1-5]College of Computing Education, University of Mindanao, Davao City, Davao del Sur, Philippines

[5]College of Engineering, Technology and Management, Cebu Technological University-Barili Campus, Cebu, Philippines

lloydlim27@gmail.com[1], pj.abing@gmail.com[2], pitzjerald.rabe@gmail.com[3], jancarlo_arroyo@umindanao.edu.ph[4], allemarjpdjca@yahoo.com[5]

ABSTRACT

Constructo application is a multi-platform single page architecture project that empowers users to experience a more efficient way of finding construction materials and comparing prices to different stores within Davao City, Philippines, utilizing the android and web versions of the system. The application enhances the capability of the store owners to showcase individual detailed products and accommodate orders regardless of the number of product materials being ordered. Moreover, the study addresses the identified issue of the general processes of acquiring product materials, which includes the difficulty in finding construction materials, making a list of ordered products, canvassing product prices, and inquiry of product information. The use of agile scrum methodology was observed in the development phase of the system. Results as to compatibility issues were provided in the latter part of the paper, revealing the overall functionalities of the implemented system applications.

Key words: Constructo app, mobile application, multiplatform application, single-page architecture, web application

1. INTRODUCTION

Over the past century, the growth of the industry has become remarkable not only because people nowadays are bound by technology but also because of the mass production of building establishments. Furthermore, industry building establishments cannot be made possible without the use of some certain construction materials, which primarily the main foundation of every man-made buildings.

In Davao City, Philippines, many people are starting to upgrade their homes as it provides a variety of good environmental feeling and a wide space works. For starters who plan to make house upgrade, inquiry on products and services, as well as the canvassing of prices for materials, are some of the most tedious tasks that need to be done. To leverage on the use of technology for the ease of doing the abovementioned task, an Android and web application that aims to help the customer in gathering information about a certain product from a certain store in a small amount of time considering the time and effort wasted when one can be improved by the use of technology is undertaken.

The study aims to eliminate some tedious processes in the acquisition of material and products from a certain store. Further, the study aims to improve the quality of services that each of the stores provides when it comes to accommodation of inquiries and services by providing a technology that everybody can use. Therefore, the main objective of the study is to provide a “Constructo” application that enables the customers to make orders and inquire about the information of building materials within the selected store. Specifically, the study aims to develop an android application using Ionic and Apache Cordova, view locations of the various registered stores in the application using Google Map API, update the store products in real-time using the Pusher Notification API, search and compare prices from the different registered stores in the application using Laravel Query Builder of the Laravel Framework, order the construction materials online using Ajax Request and Pusher Notification API, and block flagged fraud users using Laravel Query Builder, among others.

2. RELATED SYSTEMS

2.1 MarketWatch

Market Watch for Android is an app that delivers fresh business news and other marketing issues; it's like a marketing newspaper in your android device. MarketWatch is built with the Ionic framework regarding the UI/UX design structure and the front-end structure [1].

2.2 Bazaar

Ambazzar application is one of the leading online web-based system combining with a mobile application that supplies construction materials in Bihar & Jharkhand. With the use of the online system and mobile application, the customer from the said country can easily pick certain product materials along with the material price. Also, it offers a wide range of variety of products at its best price. Also, the system provides some construction services, which include suggestions of the possible plumbing contractor, building contractor, and construction equipment provider, among others [2].

2.3 Kumukutu Shop

A combination of web-based system and mobile application that brings manufacturers, vendors, and buyers of building and construction materials altogether into one platform that enables them to do trading and buying of certain materials. Thus it shows all variety of materials that each of the stores provides at its original price [3].

2.4 ConstroBazaar

ConstroBazaar is one of the largest online shops for sellers and buyers who seek to buy high-quality construction materials, interior products, and construction services. It also functions as an online purchase manager who primarily identifies only qualified sellers that matches the customer's preferences of buying the right material for upgrades or reforms. Moreover, ConstroBazaar also helps customers to make a profound deal to the sellers considering the best price they can give out from the selected materials [4].

2.5 Hines Supply

Hines Supply application is an android based application that helps the professional contractor to simply track job site deliveries, check billing information, sign up for events promotions, and more, which considerably saves a lot of time and money [5].

2.6 Hindustan Steel Suppliers

Hindustan Steel Suppliers is one of the leading building construction materials store in Bangalore and authorized wholesale distributors & retailers of TMT Steel, Cement, Sheets, Moulding Mat & Hardware Supplies. The company aims to provide an online marketplace together with the State of the art Technology (with a most secure website on the internet) that will let the customer to conveniently select various materials at their choice [6].

3. METHODOLOGY

3.1 Data Gathering

Series of interviews with different identified users of the system within Davao City, Philippines, surveys to the possible users of the project, and some sort of study reviews were done to acquire essential details for the system development.

3.2 Development Tools

3.2.1 Apache Cordova

To provide an Android version of the application, the use of Apache Cordova, a mobile application development framework created by Nitoby, was observed. In the year 2011, it was purchased by Adobe Systems and rebranded as PhoneGap. The Apache Cordova is used as a tool to compile web components into an Android APK file by coding the intended Cordova command for generating the APK file.

3.2.2 Laravel Framework

The connections and structural design for the application's web version are implemented and developed using the Laravel Framework. Laravel is a PHP framework developed by Taylor Otwell. The use of the Laravel and Vue framework were instrumental to the development of a single page architecture of the application. The study used the Laravel Framework to prevent repetitive coding by using the controllers. This also provides an organized structural connection of components such as creating connections from a controller to a view that provides an interface with functions, as well as connections from middleware to controller to provide an HTTP filtering mechanism, and many more.

3.2.3 VueJS Framework

To provide the single-page architecture of the application and to realize the organized data structuring from HTML templates to script functions, the implementation of the VueJS Framework was observed as it is used as a front end Vue layer JavaScript Framework. The VueJS is instrumental in creating HTML templates, HTML tags, and creating functions by coding it in JavaScript syntax and binding the two components virtually.

3.2.4 AJAX

Asynchronous JavaScript and XML are used to create asynchronous transferring of data between the client-side and the server-side of the application. Sending and receiving data from the server to the client is made possible using AJAX.

3.2.5 Visual Studio Code

The Visual Studio Code editor is instrumental in the development of the study. Visual Studio Code is a source code editor and is developed by Microsoft for Cross-platform desktop operating systems. It includes debugging features, built-in Git control, syntax highlighting, downloadable snippets, and code refactoring. In the development of the application, the Visual Studio Code is used as a default text editor for coding HTML, PHP, JavaScript, and Angular Typescript syntax. The Visual Studio Code provides intelligent syntax among the downloaded snippets, which are a handy feature as it provides context awareness for a certain code.

3.2.6 Ionic Framework

The Ionic Framework was used to provide an interactive android interface of the application. The ionic framework is an HTML 5 framework developed by the Ionic team and with the help of their core dependency, which is the Angular Framework from Google Company. The Ionic Framework is used in building an interactive android interface by using its prefix native Android features such as Android's native status bar, Android's native notification bar, and others. These can be done by stating a respective Cordova plugin into the application.

3.2.7 Google Maps API

The Google Maps API is used to provide a graphical map of the application wherein customers can view the store on a graphical map of Davao City. This is realized through the use of places library under the Google Maps API that grants the application to view store location. Further, this allows store owners to view the customer's graphical location by the use of Geocoder API from the Google Maps API wherein a string of address are converted as a latitude and longitude coordinates.

3.2.8 Pusher Notifications

The use of Pusher Notifications API was observed to provide real-time notifications and the real-time chat feature of the application. Pusher is a web socket API that is capable of providing full-duplex communication channels over a single TCP connection. This is realized by assigning a broadcast object to a specific Laravel controller, which has a function of retrieving or sending data and assigning a channel to the front end in the VueJS. The broadcast and the channel are bonded to work as a receiver and a sender in which the broadcast object broadcasts the message to the channel in a real-time manner.

3.3 Conceptual Framework

The methodology used in this study is the Agile Scrum methodology and are in forms of the sprint to wit: requirement analysis, analysis and design, development, debugging, and implementation. Figure 1 shows the general conceptual framework of the Constructo app. The following subsection shows the input, process, output of the functionalities found in the study.

3.3.1 IPO Model – Ordering

Figure 2 shows the IPO model for ordering system where ordering is done with the use of the web or android application. In sending orders, the data from the order list are encoded in the JSON object and are sent via Ajax request directly to the webserver. The data from the JSON object are saved to the database and real-time notification, and a PO update is sent via a web server with the use of the Pusher Notification API. The JSON object is parsed by the front-end user interface (UI) and is seen by the seller. Lastly, the seller prepares and delivers products depending on the terms of the payment process set by the customer/user.

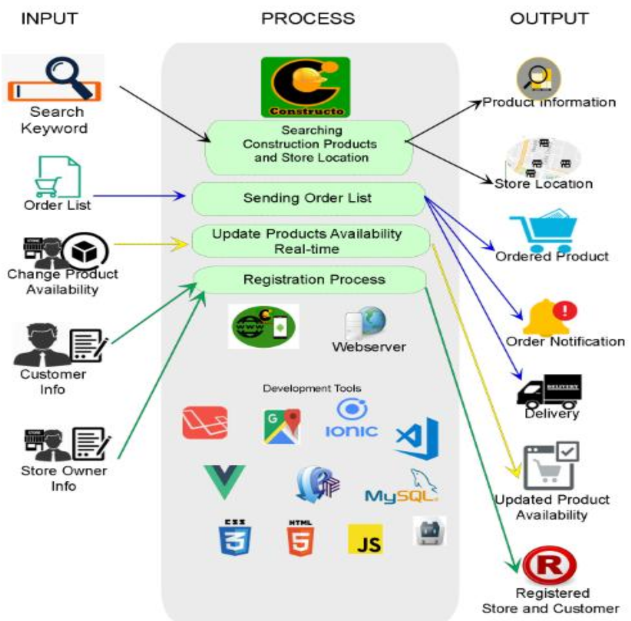

Figure 1: Conceptual framework of the system

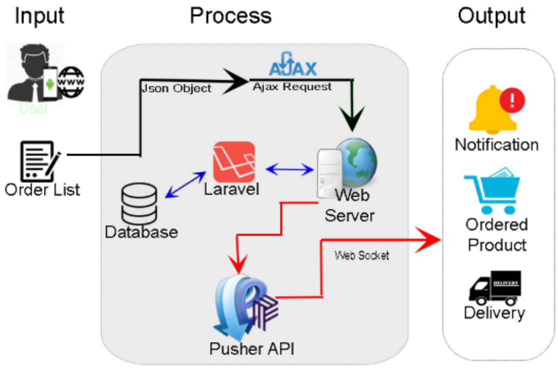

Figure 2: IPO model for ordering system

3.3.2 IPO Model – Product Search and Compare

The customer can search and compare the prices of the products from different stores in the application with the help of the Laravel Query Builder. The front-end UI is developed with a rangefinder for the product price and a search box for searching for a specific product name. When the Customer searched a product, all products with different prices from different stores, according to the searched product name, are displayed in the form of a list. Customers can now compare the product's price from different registered stores.

All data coming from the customer's side are encoded to JSON data types and are sent through Ajax request. The Laravel Query Builder processes data for validation. After validating the data and proved to be legitimate, data is saved to the web server that is part of the application's back-end side. Laravel Query Builder implements a convenient and fluent interface for running database queries. It tends to be utilized to perform most database tasks in your application and works on all supported database systems.

When searching for a product, the searched product name will be filtered by the Laravel Query Builder. Then it will fetch the products from the database according to the searched product name. Data from the rangefinder filter the products according to their prices. It will select the products if the prices are less than or equal to the data value of the rangefinder. The response from the application's back-end side is handled by the front-end side of the application. The graphical representation of the process is shown in Figure 3.

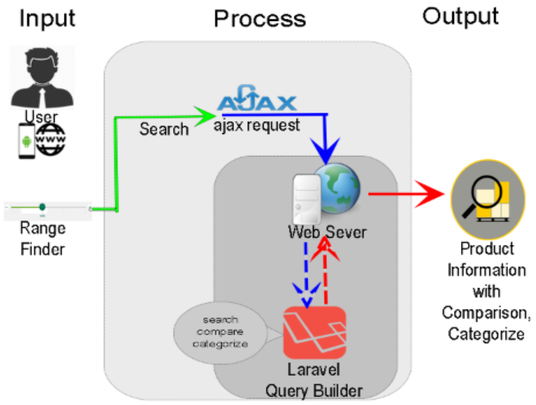

Figure 3: IPO model for product search

3.3.3 IPO Model – Real-Time Update of Products Availability

In the process of making a real-time update of the products, stores are given the privilege to update the product with the use of Pusher Notification API. Pusher is a web socket API that provides full-duplex communication channels. Product status is updated by updating the product database. The request is handled by Ajax, Pusher PHP code, WebSocket, and Pusher JavaScript code to update the web and android applications in real-time, as shown in Figure 4.

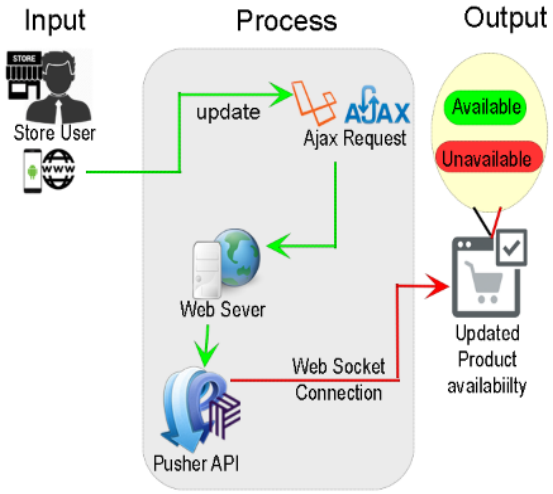

Figure 4: IPO model for the real-time update of products process

3.3.4 IPO Model – Google Location Finder

The use of Google Map API was observed to view the location of the different registered stores in the application. The Google Map API allows the app to display store markers that represent the store's location according to the address of the different registered stores. By clicking the store marker on the map, a small info window is displayed with the visit button of the store. After the customer sends the order, the store can also locate the customer on the map base on the address set or given by the customer. The IPO model for the store location process is shown in Figure 5.

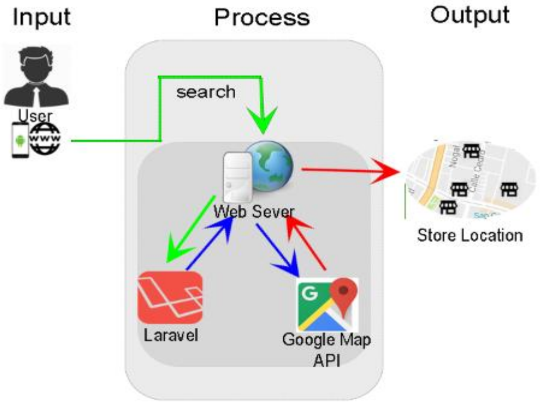

Figure 5: IPO model for store location using google map

3.3.5 IPO Model – Hybrid Technology

The combination of two or more elements to make a new variety of an element is called a hybrid implementation. In the study, a hybrid application is developed by structuring web components that can be compiled into an Android application. The application was developed through web programming codes with native android wrappers such as Cordova and PhoneGap.

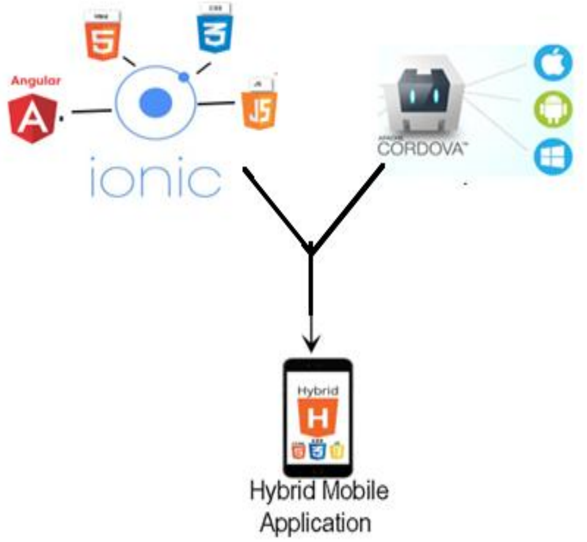

Figure 6: IPO model for creating hybrid technology

Hybrid technology for the Android application is developed with the use of the Ionic framework and Apache Cordova. Ionic framework is an HTML 5 framework developed by the Ionic team. Ionic framework is the front-end UI used to build the android application of the study, which can be developed through HTML, CSS, and JavaScript. Instead of using pure JavaScript, the Ionic framework is used along with the AngularJs Framework to achieve the core functionality of the application. Further, the tool used to compile the web components into an Android APK file is the Apache Cordova created by Nitoby, which generate APK file using codes of Cordova command. The IPO model for the creation of hybrid technology is shown in Figure 6.

3.4 Single Page Architecture

Figure 7 shows an illustration of how the Single Page Architecture is structured and how it works in most web applications such as Facebook, Trivago, Lazada, and other web application wherein it does not refresh the whole page upon navigation, but instead, it only refreshes the necessary content of the page. The process saves memory consumption and data network at the application's end, which enhances user experience. The Single Page Architecture can only be implemented by acquiring the VueJS JavaScript Framework. A modular router is provided by the VueJS JavaScript Framework that handles the shuffling of the templates depending on the page navigated by the user as each template represents a page that contains style and unique functionalities that makes each of them a page in a web application. The Single Page Architecture is used to attain data flexibility and fluidity of navigation that enhances user experience. The architecture utilizes the VueJS Framework for creating HTML templates and is loaded through Vue's modular router.

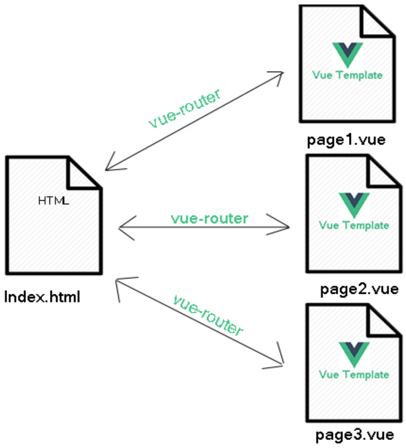

Figure 7: Single page architecture diagram

3.4 Use Case Diagram

Figure 8 shows the use case diagram of the Constructo application and the level of accessibility for each of the identified users of the system. System functionality includes add and update products, receive orders from the customer, view orders from the customer, receive notification from the admin, search/view products, create a list of wanted products, and send an ordered list to a certain store, among others.

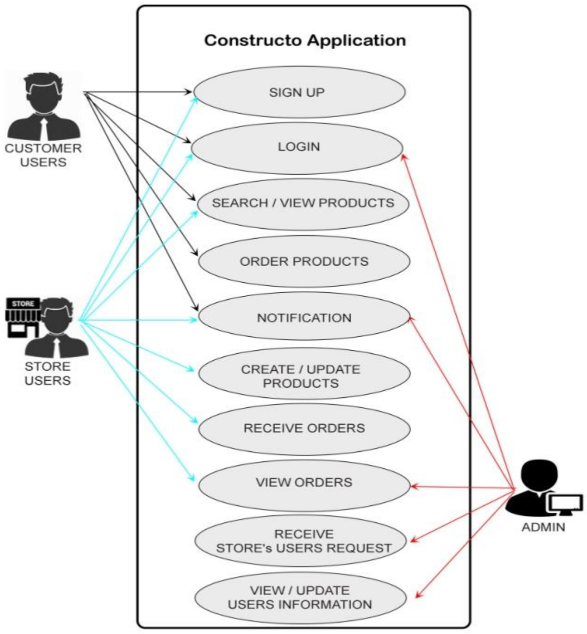

Figure 8: Use case diagram of the study

4. RESULTS AND DISCUSSION

4.1 Black Box Testing

This study utilized some level of applicable testing methods, which includes RAM and CPU tests, web browser test, android platform test, and functionality test. Tables 1-13 shows the result of the abovementioned tests conducted.

Table 1: RAM and CPU test results

Task	Remarks	Result
1.0 GHz processor, 1.0GB RAM, 1.0GB Internal Storage, 720 x 1280 px	Passed	Runs Smoothly
1.3 GHz processor, 1.0GB RAM, 8.0GB Internal Storage, 720 x 1280 px	Passed	Runs Smoothly
3.4 GHz processor, 8.0GB RAM, 65.0GB Internal Storage, 720 x 1280 px	Passed	Runs better than the previous results

Table 2: Web browsers test results

Task	Remarks	Result
Internet Explorer 8	Failed	Browser failed to load the main interface of the system
Mozilla Firefox 65.0	Passed	Browser successfully loaded the web system
Google Chrome 71.0.3578.98	Passed	Browser successfully loaded the web system
Opera Mini 12.15	Failed	Browser failed to load the main interface of the system

Table 3: Android OS versions test results

Android Version	Remarks	Result
4.4 (kitkat)	Passed	Installed & Run Successfully
5.0 (Lollipop)	Passed	Installed & Run Successfully
6.0 (Marshmallow)	Passed	Installed & Run Successfully
7.0 (Nougat)	Passed	Installed & Run Successfully
8.0 (Oreo)	Passed	Installed & Run Successfully

Table 4: Admin dashboard functionality test result

Task	Expected Output	Result
Click "Dashboard"	Shows the number of a pending request, registered customer, registered store, product categories, validated records, all transactions.	Passed
Click "Number of all pending request"	Will be redirected to the request panel.	Passed
Click "Registered store and customer card"	Will be redirected to the user's page application under admin	Passed
Click "Product Categories"	Will be redirected to the product category module.	Passed
Click "Records"	Will be redirected to the admin's record module	Passed

Table 5: Admin request functionality test result

Task	Expected Output	Result
Select list under sort by request type drop-down list and click the search button	Shows all the filtered request type (Registration, Subscription, and Concern)	Passed

Table 6: Admin manage request module functionality test result

Task	Expected Output	Result
Click "Manage Request" under the registration request type.	Will show the store's name, location, phone #, email address.	Passed
Click "Accept Request" under manage request form	Will show [store's names] request accepted	Passed
Click the "Reject with Concern Request" button under manage request form	Will show the store's information and concern text field	Passed
Click the "Reject Request" button under manage request form	Will show [store's name] request rejected	Passed

Table 7: Admin records module functionality test result

Task	Expected Output	Result
Click "Record" on side menubar	.Shows the store's record regarding request type, request status, date accommodated and who validate the request	Passed
Click the "Rejected" button	Shows the rejected concern message	Passed

Table 8: Admin product category functionality test result

Task	Expected Output	Result
Click "Product Category" on the side menu bar	Shows all the valid product categories	Passed
Click "Search button"	Shows the searched product	Passed
Click the "Add Category" button	Shows the add category form	Passed
Click the "Edit" button	Shows the edit product category form	Passed

Table 9: Admin user management functionality test result

Task	Expected Output	Result
Click "Users" under admin navbar	Shows all the number of customer and store's information and their location in google map	Passed
Type on the search field and click the "Search icon" button.	Shows all the searched user's information	Passed
Click the "Customer" and "Store" Tab	Shows the information regarding the store or customer	Passed

Table 10: Store dashboard functionality test result

Task	Expected Output	Result
Click "Dashboard" At side navbar	Shows the number of all orders, products and product categories cards; also will show the top 3 best-selling products and top 3 buyers;	Passed
Click the number of all order card	Will be redirected to the order module	Passed
Click the number of all product card	Will be redirected to the product module	Passed
Click the number of all product categories	Will be redirected to the product category module	Passed

Table 11: Store order module functionality test result

Task	Expected Output	Result
Search orders on the search field and select payment term, status and date for filtering	Shows all the searched orders with the corresponding order ID, the customer who orders, date ordered, due date, payment term, and the order status	Passed
Click certain orders	Will be redirected to another page containing ordered items, order info, and customer info	Passed

Table 12: Store order module functionality test result

Task	Expected Output	Result
Search orders on the search field and select payment term, status and date for filtering	Shows all the searched orders with the corresponding order ID, the customer who orders, date ordered, due date, payment term, and the order status	Passed
Click certain orders	Will be redirected to another page containing ordered items, order info, and customer info	Passed

Table 13: Store products module functionality test result

Task	Expected Output	Result
Click "Products" On side menubar	Shows all the products of the store	Passed
Search product on the search field and click "search" button	Shows the searched product	Passed
Click "Edit product" button on a certain product	Shows the edit product form for editing products	Passed
Click the "Set to Unavailable" button on a certain product	The product status will turn into unavailable	Passed
Click "Remove Product" button on a certain product	The product selected will be removed from the list of store products	Passed
Click Add Product	Will show the adding form for adding products	Passed

Table 14: Store product category functionality test result

Task	Expected Output	Result
Click "set up payment terms"	Shows the payment terms that can be set by the store	Passed
Click "Change password"	Shows the edit form for changing password	Passed

Table 15: Store product category functionality test result

Task	Expected Output	Result
Click "Product Category" (side navbar)	Shows all the available products set and approved by the admin together with some functional buttons	Passed
Search product category on	Shows the searched product	Passed

the search field and click the search button	category	
Click the "Add category" button	Shows the add category form containing the list of approved and available product category	Passed
Click "Request Category" Button	Shows the request category form	Passed
Click "Remove Category" button from a certain product	Remove the selected product category from the store's list of the product category	Passed

Table 16: Store notification functionality test result

Task	Expected Output	Result
Click "Notification" (side navbar)	Shows all the admin replies in response to the store request with the following information: Request type, Request status, Accommodated by and the date of the request was validated	Passed
Click the "Reject" button (if available)	Shows the message that corresponds to why the admin decline the request	Passed

Table 17: Store account settings functionality test result

Task	Expected Output	Result
Click "My Account" (on navbar)	Shows the store's information	Passed
Click "choose profile" button, select the proper image and click "save profile picture" button	Image profile selected will be saved	Passed

Table 18: Customer menu functionality test result

Task	Expected Output	Result
Log-in as customer	Shows the main interface of searching products together with some filtering options, which include the product's price, category, and payment terms.	Passed
Search product on the search field by using the filtering options and click the search button.	Shows all the product searched containing some valuable information regarding the products store and its location	Passed
Click "View Map" button (Available after a product is searched)	Shows the satellite map of all the store location and the customer's current location	Passed
Clicked any searched product	Will redirect to the product page module	Passed
Click "My Orders" (Customer Navbar)	Will redirect the customer to his/her history of orders	Passed
Click "My Account" (Customer Navbar)	Will show the customer's personal information and current location on the google map's satellite	Passed

Table 19: Customer product module functionality test result

Task	Expected Output	Result
Click the "Order Now" button	Will show the "order now" form the contains the information regarding the selected product	Passed
Click "Add to cart" button	Will show the "Add to Cart" form containing the function button that will add the selected product to the customer's cart list	Passed
Click "Chat Now" button	Will show the compose message form containing the functional button for sending a direct message to the store	Passed
Click the "More Product" button	Will show "more product" form under the selected product from a certain store	Passed

Table 20: Customer add to cart module functionality test result

Task	Expected Output	Result
Click "Increment [+] and decrement [-] icon button	Will automatically add or subtract the given price base on the quantity set by the customer.	Passed
Click "remove icon" button	Will remove the selected products from the cart	Passed
Click "Check Out" button	Will show the "check out" form	Passed

Table 21: Customer check out module functionality test result

Task	Expected Output	Result
Select COD (Cash on delivery) on payment	Will be asked for the date of delivery before a customer can check out.	Passed
Select CUP(Cash upon pick-up) on payment option	Will be asked for the date of delivery and delivery address before a customer can check out.	Passed

A series of test results had been imparted in the form of tables shown in Tables 1-21, with the content of multiple data that supports the idea of the study's multi-platform and single-page architecture. The test results show that the system had been completely proven to be compatible with android devices with an OS version of 4.4 (Kitkat), 5.0 (Lollipop), 6.0 (Marshmallow), 7.0 (Nougat), and 8.0 (Oreo). For the web platform, the system runs smoothly when Mozilla Firefox (65.0) & Google Chrome (71.0.3578.98) were used. Further, the system successfully runs on computers with minimum specifications to wit: 1.0 GHz processor, 1.0GB RAM, 1.0GB Internal Storage PC. The functionalities of the system have been met, making the study a success.

5. CONCLUSION

The specific objectives of the study to wit: develop web and android-based application for users to create and send a purchase order of construction materials, view different registered construction store in the application via built-in google map, update products availability in a real-time manner, and provide a list of searched construction products with the corresponding price and status of availability among others, are met. The complete description of every functionality that is present in the system application is shown in the test result on Tables 1-21, supporting the claim that the system is fully operational, and the major objectives are achieved.

REFERENCES

[1] Jones D (2018). MarketWatch. Retrieved September 2018, from https://play.google.com/store/apps/details?id=com.marketwatch&hl=en&_ga=2.54303150.78793495.1537803029-714224591.1537803029

[2] PI Systems Private Limited (2018). Ambazaar. Retrieved September 2018, from

https://play.google.com/store/apps/details?id=com.pisystem.material

[3] Bamwine N (2018). Kumukuto Shop. Retrieved September 2018, from https://play.google.com/store/apps/details?id=com.kumukutu.mobilestore

[4] ConstroBazaar Pvt (2018). ConstroBazaar: Your Online Purchase Manager. Retrieved September 2018, from https://play.google.com/store/apps/details?id=pkg.cb

[5] USS LBM Holdings, LLC (2018). Hines Supply.Retrieved September 2018, from https://play.google.com/store/apps/details?id=com.uslbm.hines

[6] Appswiz MMA (2018).Hindustan Steel Suppliers. Retrieved September 2018, from https://play.google.com/store/apps/details?id=com.appswiz.hindustansteelsuppliers