



Licenciatura em Sistemas e Tecnologias da Informação

**KARIAKOO GOES ONLINE: Placing Traditional Tanzanian Market
on an Online Marketplace Platform**

Projeto Final de Licenciatura

Elaborado por LYINDER NELSON SWALE

Aluno nº 20182295

Orientador: Professor Doutor Alexandre Humberto dos Santos Barão

Barcarena

Junho de 2020

Atlântica

Licenciatura em Sistemas e Tecnologias da Informação

**KARIAKOO GOES ONLINE: Placing Traditional Tanzanian Market
on an Online Marketplace Platform**

Projeto Final de Licenciatura

Elaborado por LYINDER NELSON SWALE

Aluno nº 20182295

Orientador: Professor Doutor Alexandre Humberto dos Santos Barão

Barcarena

Junho de 2020

O autor é o único responsável pelas ideias expressas neste relatório

Acknowledgements

First of all, I thank the most high God for giving me endless second chances and for directing my path always.

I'm deeply indebted to the love of my life KG for believing in me and for funding my education. His consistency, moral support and above all an everlasting love have given me the strength I never thought I had, so thank you babe.

I'm also grateful to my great and unforgettable professors. My supervisor Professor Alexandre Barão for helping and leading me to accomplish my dream by his tireless and continuous support, thoughtful comments and recommendations on this dissertation. Professor Antonio Aguiar for a great foundation and positive push towards my goal and the head of our faculty Professor Carla Silva for her empowerment.

I am also thankful to Universidade Atlântica and all its staff for all the considerate guidance. To conclude, I cannot forget to thank my family and friends for all the unconditional support in this very intense academic year.

Resumo

KARIAKOO GOES ONLINE: Placing a Traditional Tanzanian Market on an Online Marketplace Platform

O mercado de *Kariakoo* é o maior e mais típico exemplo de um mercado tradicional na *Tanzânia* localizado na antiga capital, *Dar es Salaam* juntando mais de 5000 comerciantes que conduzem os seus negócios por grosso, e, retalhistas com uma vasta gama de produtos e serviços. Os negócios no mercado são conduzidos em lojas físicas, enquanto os compradores têm de ir fisicamente ao mercado para fazerem as suas compras. A revolução nas tecnologias da informação e da comunicação mudou a forma como as pessoas conduzem os negócios, permitindo a integração dos processos de negócio dos mercados tradicionais e traduzindo-os no ambiente virtual, nomeadamente no mercado eletrónico. Este artigo discute o conceito de e-business e os diferentes tipos, incluindo o e-marketplace. Consequentemente é desenvolvido um artefacto de e-marketplace chamado *Tznini Market* usando a metodologia de *Design Research Science*. O mercado é desenvolvido com o propósito de ser usado por vendedores e compradores do mercado de *Kariakoo*.

Palavras-chave: *Kariakoo market, TZNINI, Online marketplace, Electronic business, Network economy, Client-server application, Responsive application, Platform business model, WordPress, WooCommerce, Dokan*

Abstract

KARIAKOO GOES ONLINE: Placing a Traditional Tanzanian Market on an Online Marketplace Platform

Kariakoo market is the largest and a typical example of a traditional market in Tanzania located in the former capital city, Dar es Salaam with a collection of over 5000 merchants conducting their businesses wholesale and retail with a wide range of products and services. Business at the market is conducted brick-and mortar whereas buyers have to go physically to the market for purchases. The revolution in information and communication technology has changed the way how people conduct business by enabling traditional markets business processes integration and translates them into the virtual environment namely e-marketplace. This paper discusses the concept of e-business and the different types including e-marketplace. An e-marketplace artifact named tznini market is then developed using design science research methodology. The marketplace is developed for the purpose of being used by Kariakoo market vendors and buyers.

Keywords: Kariakoo market, TZNINI, Online marketplace, Electronic business, Network economy, Client-server application, Responsive application, Platform business model, WordPress, WooCommerce, Dokan

Index

Acknowledgements.....	iii
Resumo	iv
Abstract.....	v
Index	vi
Index of Figures	xi
Index of Tables	xii
Abbreviations and Acronyms	xiii
1. Introduction.....	1
1.1. Context and Motivation	1
1.2. Problem Definition	2
1.3. Research Goals	3
1.4. Research Methodology	3
1.5. Achieved Results	6
1.6. Document Structure	6
2. Literature Review	7
2.1. Electronic Business.....	7
2.1.1. Concept & Definition	7
2.1.2. Types of E-business and their application	8
2.1.2.1. E-procurement	9
2.1.2.2. E-government	10
2.1.2.3. E-commerce	10

2.2.	Electronic Marketplace	15
2.2.1.	Concept and definition.....	15
2.2.2.	Types of Electronic Marketplaces	15
2.2.3.	Benefits of Electronic Marketplaces.....	16
2.3.	Electronic Marketplace Building Approaches.....	17
2.3.1.	Custom development or ad hoc programming.....	17
2.3.2.	Off-the-shelf solutions	18
2.3.2.1.	Software as a Service (SaaS) solution	18
2.3.2.2.	Application Programming Interface	19
2.3.2.3.	Platform as a Service	20
2.3.2.4.	Content Management System.....	21
2.4.	WordPress as the Electronic Marketplace building solution.....	22
2.4.1.	Web application	22
2.4.1.1.	Understanding a web application.....	22
2.4.1.2.	Web app architecture (3-tier client-server model).....	23
2.4.1.3.	Benefits and uses of client-server model	24
2.4.2.	WordPress.....	25
2.4.2.1.	Concept and features.....	25
2.4.2.2.	WordPress as a Framework vs Model-View-Frameworks.....	26
2.4.2.3.	WordPress technologies (Themes, Plugins and Databases)	28
2.4.2.3.1.	Presentation layer (Themes)	28
2.4.2.3.2.	Business logic layer (Plugins)	30

2.4.2.3.3.	Data layer (WordPress Databases)	32
2.4.3.	WordPress E-commerce	35
2.4.3.1.	Shopping cart plugins	35
2.4.3.2.	Membership plugins	37
2.4.4.	WordPress Electronic Marketplace	38
2.5.	Web hosting	41
2.5.1.	Web hosting fundamentals.....	41
2.5.1.1.	Web hosting definition	41
2.5.1.2.	Types of Web Hosting	42
2.5.1.3.	cPanel Service.....	44
2.5.1.4.	SSL Secure server.....	44
2.6.	WordPress Security	45
2.6.1.	Basic security measures.....	45
2.6.2.	Hardening WordPress installs.....	45
2.6.3.	Backing up	49
2.6.4.	Scanning	50
2.6.5.	Security plugins	51
3.	Architecture Framework.....	53
3.1.	Domain Model	53
3.2.	Use Cases.....	54
3.3.	Database Structure	57
3.3.1.	Tznini Database Structure Diagram.....	57

3.3.2.	Tznini Database Tables Overview	58
4.	System Prototype	60
4.1.	General description	60
4.2.	User Interface.....	60
4.2.1.	Interface Structure.....	60
4.2.2.	User Interface Screenshots.....	63
4.2.2.1.	Homepage	63
4.2.2.2.	Products post page	63
4.2.2.3.	Vendors page	64
4.2.2.4.	Information about the platform.....	65
4.2.2.5.	Contacts page.....	65
4.2.2.6.	My account page.....	66
4.2.2.7.	Admin/vendor dashboard.....	67
4.2.2.8.	Dashboard settings.....	67
4.2.2.9.	User account page.....	68
4.2.2.10.	Product view	69
4.2.2.11.	Cart View	70
4.2.2.12.	Checkout	70
4.2.2.13.	Order details.....	71
5.	Validation Scenario.....	72
5.1.	Scenario Purpose, Stakeholders and Equipment.....	72
5.2.	Validation Instance	73

6. Conclusion	75
7. Bibliography	77
Annex.....	82
A. Back-Office UI Screenshots	82
B. Database Tables	85
C. Responsive Design Screenshots	87
D. Tznini Digital Marketing Plan	Erro! Marcador não definido.
Appendix.....	88
A. Design Science Research Reference Paper.....	Erro! Marcador não definido.

Index of Figures

Figure 1 Current Kariakoo Market Business Process	2
Figure 2 Design Science Research Process	4
Figure 3 3-Tier Client-Server model	24
Figure 4 Model-View-Controller (MVC) Model.....	27
Figure 5 Responsive Web Design Example	29
Figure 6 Custom WordPress blog database structure	33
Figure 7 WooCommerce multivendor plugins stats	38
Figure 8 tznini domain model.....	53
Figure 9 Vendor on tznini Marketplace platform	55
Figure 10 Buyer on tznini Marketplace platform	56
Figure 11 tznini database structure	58
Figure 12 tznini interface structure	60
Figure 13 tznini homepage UI	63
Figure 14 tznini products post page.....	64
Figure 15 tznini information about us page	65
Figure 16 tznini contacts details page.....	66
Figure 17 tznini user(my account) account page	66
Figure 18 tznini admin/vendor dashboard page.....	67
Figure 19 tznini dashboard settings page.....	68
Figure 20 tznini user account page	69
Figure 21 tznini single product view page.....	69

Figure 22 tznini cart view page.....	70
Figure 23 tznini checkout page.....	71
Figure 24 tznini order details view	71

Index of Tables

Table 1 Types of E-business (Coppel, 2000).....	9
Table 2 Traditional Commerce vs E-commerce (Vidyapeeth, 2010).....	12
Table 3 Model-View Framework vs. WordPress (Messenlehner & Coleman , 2019)...	27
Table 4 Best WordPress plugins and their functions.....	31
Table 5 Custom WordPress blog database tables overview	33
Table 6 WordPress E-commerce solutions comparisons (Schäferhoff, 2018).....	36
Table 7 Multivendor Plugins features and prices	39
Table 8 Types of Web Hosting Servers (Bodo, 2003).....	43
Table 9 tznini database tables overview	58

Abbreviations and Acronyms

KKOO – *Kariakoo*

API – *Application Programming Interface*

CMS – *Content Management System*

EM – *Electronic Marketplace*

HTTP – *Hyper Text Transfer Protocol*

PaaS – *Platform as a Service*

SaaS – *Software as a Service*

SEO – *Search Engine Optimization*

MVC – *Model-View-Controller*

SoC – *Separation of Concerns*

HTML – *Hypertext Markup Language*

CSS – *Cascade Style Sheet*

PHP – *Hypertext Preprocessor*

VPS – *Virtual Private Server*

RAM – *Random Access Memory*

CPU – *Control Processing Unit*

SSL – *Secure Socket Layer*

URL – *Uniform Resource Locator*

UI – *User Interface*

SQL – *Structured Query Language*

APP – Application

FTP – File Transfer Protocol

MTP – Mail Transfer Protocol

1. Introduction

1.1.Context and Motivation

1.1.1. Understanding Kariakoo market and its current situation

Kariakoo market is the largest market in Tanzania located in the former capital city, Dar es Salaam. The market is a typical example of a traditional market in Tanzania and has been in operation since 1923. It has a collection of over 5000 merchants conducting their businesses wholesale and retail with a wide range of products and services. This market also serves as the business center for wholesalers and exporters from other parts in the country bringing their produce/products to sell to merchants at the market or for exports. Importers also bring their products from other countries to sell mostly wholesale. Merchants from other parts of the country as well as from other countries also come to buy products wholesale to go and resell.

Business at the market is conducted brick-and mortar whereas buyers have to go physically to the market for purchases. The sellers' exposure of their businesses is as good as the position of its store at the market and the number of buyers that see the store. The market has a multitude of stores, merchants that are not formally arranged and some in hidden spaces that it can be difficult for the buyers to find what they are looking for in the market.

Also, the wholesalers (importers and exporters) tend to have a difficult time connecting to the market merchants especially if/when they are new in the business and unfamiliar with the market. This causes several drawbacks in the business process like for example the so-called agents to manipulate prices for both parties and lack of transparency.

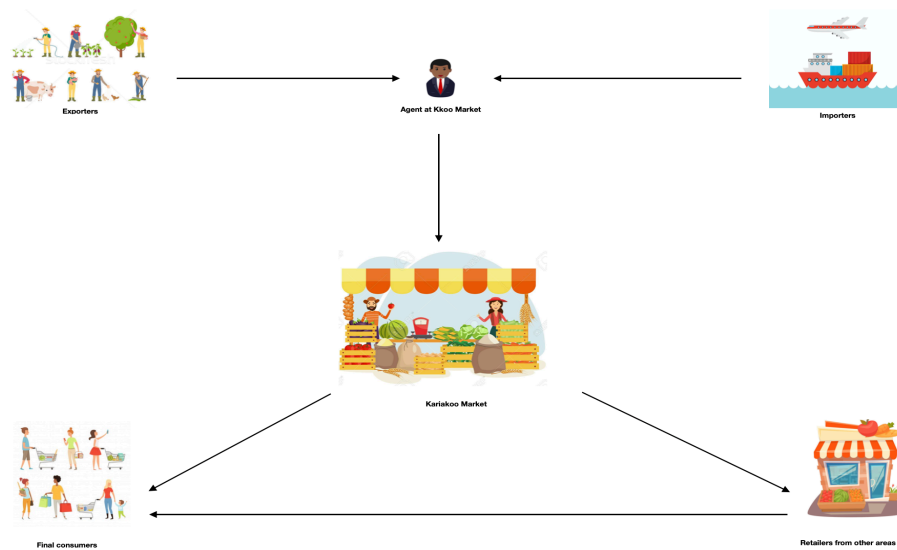


Figure 1 Current Kariakoo Market Business Process

1.2. Problem Definition

E-commerce can play a big part in improving the quality of conducting business in Kariakoo market. However, there are several problems that slow down its implementation. Namely:

- The fact that most sellers are not aware of e-commerce benefits;
- The inability for sellers to operate an online store since it is costly and requires constant maintenance;
- It is difficult to build and maintain trust between individual online vendors and buyers in terms of the quality and genuineness of the product, price manipulation, security of payment and other fraud related factors; and,
- Shipping as a challenge for sellers due to the address system in Tanzania and the cost of shipping.

1.3. Research Goals

Regarding the referred problem, several research goals (RG) were defined. Namely:

- RG 1 – Conduct a literature review, for the purpose of studying and analysing relevant related topics such as: electronic business, electronic marketplace, approaches to building an online marketplace platform, WordPress as marketplace web application solution, web hosting, marketplace marketing and marketplace security;
- RG 2 – Define an architecture framework in order to solve the problem by defining the system domain model and system main use cases (including system users);
- RG 3 – Select and analyse a third-party data model to support the system prototype;
- RG 4 – Install and test the application, which fully fits the needed features;
- RG 5 – Full describe the prototype user interface; and,
- RG 6 – Apply the solution to a concrete validation scenario describing its purpose, stakeholders and equipment.

1.4. Research Methodology

This research requires designing and developing an artifact that is the marketplace platform application that will solve several issues associated with conducting business at Kariakoo market. The need for a more efficient way to conduct business at the market encouraged the development of tznini marketplace platform. Tznini marketplace platform application will be researched to find if it will prove its value by using the design science research methodology.

“Design science...creates and evaluates IT artifacts intended to solve identified organizational problems.” (Hevner et al , 2004) It involves a rigorous process to design artifacts to solve observed problems, to make research contributions, to evaluate the designs, and to communicate the results to appropriate audiences.

The design science research process includes six steps demonstrated on the diagram below:

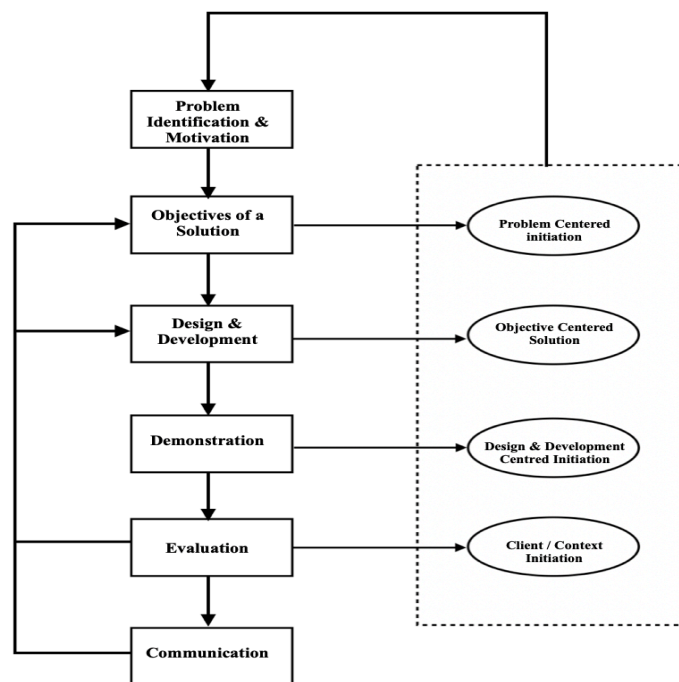


Figure 2 Design Science Research Process

Problem identification & motivation

The research problems which mainly are the need but yet difficulty to implement an e-commerce solution for Kariakoo market such as lack of knowledge and funds, addressing challenges and lack of fraud control as well as the motivation to find a solution that solves the problem best were identified on the prior sections.

Objectives of a solution

The tznini marketplace platform was designed in order to eradicate the hustle for vendors at the market to start their own e-commerce business from scratch and instead open their e-commerce stores on the platform and not having to worry about e-commerce store maintenance costs. The marketplace platform also acts as an intermediary between buyers and sellers which also preventing fraud related issues. Shipment issues also will be solved through the marketplace platform since multiple orders by the same platform buyer will be treated as the same shipment and hence save recurring shipping costs.

Design and Development

In the design and development step, a thorough literature review was conducted for the purpose of studying and analysing relevant related topics and technologies such as: electronic business, electronic marketplace, approaches to building an online marketplace platform, WordPress as marketplace web application solution, web hosting, marketplace marketing and marketplace security.

An architecture framework was then developed in order to solve the problem by defining the system domain model, system main use cases (including system users).

A third-party data model that can support the system prototype was then selected and analysed followed by an installation and testing of the application which full fits the needed features. The design and related features have been and continue to be refined based on emerging performance needs.

Demonstration

The system prototype was then illustrated by tznini marketplace interface structure diagram followed by an in-depth description of the diagram. User interface screenshots of every element of the interface structure were then presented.

Evaluation

The fully functional version of tznini e-marketplace was implemented by some of Kariakoo market vendors whereas they opened their online stores on the platform and

posted products for sale. The posted products were bought and several ordered were processed and completed. The e-marketplace platform was found to be flexible and effective in this field application.

1.5.Achieved Results

Besides literature review, main achieved results are:

- Develop a system model and architecture; and
- Implement a real-world solution (which is available online).

1.6.Document Structure

This document is organized in several chapters. In Chapter 1, an introduction is presented. In Chapter 2, the literature review is described in detail. In Chapter 3 the prototype architecture framework was described whereby the domain model, use cases and the database structure are discussed. In Chapter 4, the prototype was installed and tested as described and also, the prototype user interface was presented. In Chapter 5, a concrete validation scenario with a purpose, stakeholders and equipment description was provided. Finally, in Chapter 6, preliminary conclusion and future work are presented.

2. Literature Review

2.1. Electronic Business

2.1.1. Concept & Definition

Trade, commerce and business have always been a necessity in the footprint of human civilisation since the dawn of time. An exchange of goods, an exchange of services and an exchange of currencies. Items of value have long been utilised as a means of income and wealth.

People conduct business based on the economy trends of the time. Economy trends can be divided into three eras, pre-industrial era, during the industrial era and post-industrial era (information and technology age).

The revolution in information and communication technology has changed the way how people conduct business. Using information technology, businesses possess the potential to reach more customers, introduce new products and services quickly, and collaborate with suppliers and business partners from all over the world with less effort.

This great shift in technology has brought about a new economy, net economy. *Net economy* refers to the commercial use of digital data networks for handling information, communication, and transaction processes via various platforms (Kollmann, 2013). It is based on four technological innovations (telecommunication, information technology, media technology, and entertainment) that impact the possible ways in which information, communication and transactions are managed (Kollmann, 2013).

E-Business can be defined as the use of innovative digital information technology to sell products and services, offer and exchange information, deliver comprehensive customer support, and enable individual contact between market participants via virtual contact (Kollmann, 2013). It also includes activities for the internal management of the firm and for coordination with suppliers and other business partners as well as *electronic commerce*, or *e-commerce* (Laudon & Laudon, 2012)

E-business is usually interpreted broadly and is commonly misunderstood – mainly in relation to e-commerce (Laudon & Laudon, 2012). Although there is some

overlap, e-commerce refers to trading products and services online, and so is strictly only speaking of one aspect of e-business.

E-business is a general term that encompasses all forms of using digital information and communication technologies to support and optimize business processes. In contrast, e-commerce describes only the online trading of products and services and is therefore only a subsection of e-business.

The core components of e-business are *information, communication, and transaction*. Business partners use digital networks (i.e. public or private communication networks) to conduct business processes using innovative technologies to improve efficiency.

2.1.2. Types of E-business and their application

E-business types can be classified according to the market participants involved namely *business, consumers* and the *government*. These market participants can be divided into three distinct categories:

- i. Business (B)
- ii. Consumer (C)
- iii. Government (G)

The market participants can play the role of either buyer or service provider within the market meaning that there are nine (9) types of e-business that can be derived depending on the combination i.e. *Government-to-Government (G2G), Government-to-Business (G2B), Government-to-Consumer (G2C), Business-to-Government (B2G), Business-to-Business (B2B), Business-to-Consumer (B2C), Consumer-to-Government (C2G), Consumer-to-Business (C2B) and Consumer-to-Consumer (C2C)*.

These types of e-business, respective of their applications can be further classified into E-procurement, E-government which comprises of G2G, G2B, G2C, C2G

and B2G and E-commerce (and E-marketplaces) which comprises of B2C, B2B, C2C and C2B.

Table 1 Types of E-business (Coppel, 2000)

	Government	Business	Consumer
Government	G2G Co-ordination	G2B Information	G2C Information
Business	B2G E-procurement	B2B E-commerce, E-marketplace	G2C E-commerce, E-marketplace
Consumer	C2G Tax compliance	C2B Price comparison, E-marketplace	C2C E-commerce, E-marketplace

The matrix table above (Table 2.1) shows the different types of e-business in relation to the market participants and the respective applications (Coppel, 2000).

2.1.2.1. E-procurement

E-procurement includes sourcing, negotiations with suppliers, and R&D co-ordinating taking place on the Internet and electronic market (Yen & Ng, 2003). Croom & Johnston (2003) defined e-procurement as the mirroring of procurement activities in the Internet, while Presutti (2003) defined it as all technologies which facilitate buying using the Internet. There is a variety of e-procurement classifications, but common to all is the fact that e-procurement is composed of different applications (Knudsen, 2003).

2.1.2.2. E-government

E-government also called Electronic Government, Digital Government, Electronic Governance, and similar names can be defined as the use by government agencies of information technologies that have the ability to *transform relations* with citizens, businesses, and other arms of government (Grönlund & Horan, 2005).

It is a type of e-business that emerged in the late 1990's (Grönlund & Horan, 2005). Co-ordination by the government to the government employees (G2G), information provision to the public (G2B and G2C), the government tendering through e-procurement (B2G) and the public tax compliance (C2G) (Coppel, 2000) are a few applications of e-business in relation to the government.

WorldBank 2002 best describes the benefits of e-government as *“better delivery of government services to citizens, improved interactions with business and industry, citizen empowerment through access to information, or more efficient government management. The resulting benefits can be less corruption, increased transparency, greater convenience, revenue growth, and/or cost reductions”*

For the purpose of this study, the focus will be directed to e-commerce and e-marketplaces side of e-business.

2.1.2.3. E-commerce

E-commerce is a subset of e-business which can be defined as the process of buying and selling goods and services electronically or using the internet. Other than buying and selling, many people use Internet as a source of information to compare prices or look at the latest products on offer before making a purchase online or at a traditional store (Khan, 2016).

It enables businesses to be more efficient in their internal operations and more responsive to the needs and expectations of their customers. E-commerce technologies

enable enterprises to exchange information instantaneously, eliminate paperwork, and advertise their products and services to the global market.

Any size business can have an e-commerce strategy. Many businesses have become extremely profitable through online sales. Large companies can reduce sales and stocking costs by selling online (Vidyapeeth 2006).

E-business types in e-commerce are business-to-consumer (B2C), business-to-business (B2B), consumer-to-consumer (C2C), consumer-to-business (C2B).

B2B is the exchange of products, services or information between business entities. B2B activities includes *Direct selling and support to business*, *E-procurement* where a purchasing agent can shop for supplies from vendors, request proposals, and, in some cases, bid to make a purchase at a desired price and *Information sites* providing information about a particular industry for its companies and their employees.

In a B2B environment, purchase orders, invoices, inventory status, shipping logistics, and business contracts handled directly through the network result in increased speed, reduced errors, and cost savings.

B2C, this is the exchange of products, information or services between business and consumers in a retailing relationship. In B2C e-commerce, businesses sell directly to consumers.

In **C2C**, consumers interact directly with other consumers. They exchange information such as *Expert knowledge* where one person asks a question about anything and gets an e-mail reply from the community of other individuals. There is also an exchange of goods between people both with consumer auction sites such as e-bay.

E-Commerce features

The following are the features of e-commerce according to Vidyapeeth (2010):

- *Low Entry Cost:* An individual or a company can establish themselves on the Internet, and open for business, with a relatively small investment.
- *Reduces Transaction Costs:* Interacting with customers over the internet, whether to process orders or to attend to customer support, is cheaper than traditional marketing methods.
- *Access to the global market:* With a traditional business, the target market may be the local community, or, with a higher advertising budget, it may extend to neighbouring communities. The Web extends the reach of even the smallest businesses by allowing them to market products globally.
- *Online distribution:* The Web enables businesses to distribute data and software online.
- *Secure market share:* Getting a business online protects its current offline market share from being eroded by an online entrepreneur. If a business enters the e-commerce market too late, competitors who have already established a Web presence may make a successful market entry more difficult.

E-Commerce vs Traditional market

The Internet provides a flexible and dynamic marketplace to exchange goods, services, and information with consumers and business partners, it is becoming increasingly important for businesses to use the Internet to reach new markets. Although the goals and objectives of both e-commerce and traditional commerce are the same—selling products and services to generate profits—they do it quite differently.

Table 2 Traditional Commerce vs E-commerce (Vidyapeeth, 2010)

Activity	Traditional commerce	E-commerce
Product information	Magazines, fliers	Websites, online catalogues
Business communications	Regular mail, phone	E-mail

Check product availability	Phone, fax, letter	E-mail, websites
Order generation	Printed forms	E-mail, websites
Product acknowledgements	Phone, fax	E-mail, websites, and EDI
Invoice generation	Printed forms	Websites

Despite of all the benefits that e-commerce may bring to a business, there are several reasons why most SME are not able to implement the technology. Lehman and Alan et al (2017) classify these barriers into five categories namely; *organisational barriers, financial barriers, technical barriers, legal and regulatory barriers* and *behavioural barriers*.

Organisational barriers are results from the organizations attitude towards the implementation of e-commerce system (Flynn & Purchase, 2001). They consist of lack of business models, culture, organization and planning, lack of employee knowledge, lack of infrastructure, reluctance to link to other parties, time taken for implementation, lack of supplier interest, and lack of perceived need on e-commerce. According to Zaeid (2012) organizational barriers such as difficulty in changing the existing working procedure, lack of management support, organizational resistance to change, limited use of internet banking and web portals SMEs are some of the barriers for SMEs adoption on e-commerce. All these are “too difficult” factors or barriers which make e-commerce complicate to implement among SMEs (MacGregor & Vrazalic, 2005).

Financial barriers are costs required for investment, maintenance and risk of implementing the e-commerce system (Love et al., 2001). E-business technologies involve high implementation cost (Heung, 2003; Love et al., 2001; Zaeid, 2012). In addition, lack of financial infrastructure (Zaeid, 2012), inability to develop return on investment (Flynn & Purchase, 2001), uncertainty of payment methods (Organization for

Economic Cooperation and Development, 2004) will influence businesses decision to implement the e-commerce as well.

One of the main issues in technical barriers is the type of technologies or software adopted does not match with the current organizational business requirements (Love et al., 2001). Incompatible with technologies in business will affect business loss and increase the operational costs.

Lack of education and knowledge about e-commerce (Darch & Lucas, 2002; Duan , et al., 2002) has created the attitude of distrust towards the information technology industry (Bode & Burn, 2002). Due to increasing innovation and new technologies, e-commerce implementations have become complicated (Kaynak, Tatoglu, & Kula, 2005; Zaeid, 2012)

Legal and Regulatory Barriers occurs when consumers have the lack of ‘transactional and institutional trust’ that related to the weak rule of laws that was a major impediment to e-commerce (Efendioglu & Yip, 2004). Countries that do not concern on the computer crime or secure online transaction will prevent organization to adopt e-commerce.

Behavioural barriers concentrate on issues such as fear that jobs would be lost, reluctance to make changes in working habits, the requirement to undertake additional training and skill development, and the degree of uncertainty that technology instils in people (Love, Irani, Li, Cheng, & Tse, 2001).

With the barriers comes an opportunity of inter-organisational information system (Bakos, 1991), an electronic procurement solution (Segev & Beam, 1999), a medium (Schmid & Stanoevska-Slabeva, 1998) , a meeting point (Kaplan & Sawhney, 2000), an intermediary (Dai & Kauffman, 2000) which has the ability to minimise the barriers and allow SME and individuals to implement e-commerce. This solution is known as Electronic marketplace or e-marketplace (EM).

Is an EM an inter-organisational information system (Bakos, 1991), an electronic procurement solution (Segev & Beam, 1999), a medium (Schmid & Stanoevska-Slabeva, 1998), a meeting point (Kaplan & Sawhney, 2000), an intermediary (Dai and Kauffman, 2000) or just a listing (Bradley & Peters, 1997)? What is it really?

2.2. Electronic Marketplace

2.2.1. Concept and definition

Information technology enables traditional markets business processes integration and translates them into the virtual environment, e-marketplace (Kambil & Van Heck, 2002).

An electronic marketplace (also known as online marketplace) provide a single, digital marketplace based on Internet technology for many different buyers and sellers. They are industry owned or operate as independent intermediaries between buyers and sellers. Electronic marketplaces generate revenue from purchase and sale transactions and other services provided to clients. Participants in e-marketplaces can establish prices through online negotiations, auctions, or requests for quotations, or they can use fixed prices.

An online marketplace can be defined as an *e-commerce* web or mobile application where third-party companies can sell their products or services to consumers. All of the transactions are processed through the EM owner.

2.2.2. Types of Electronic Marketplaces

There are two types of e-marketplaces, vertical e-marketplace and horizontal e-marketplaces.

Vertical EMs are industry specific (Korper & Ellis, 2000), they aggregate supply or demand in vertical industries. Vertical marketplaces require a good deal of industry knowledge. They optimise buyer–seller relationships in a specific industry (Grieger, 2003). A few examples of well-known vertical marketplaces are Airbnb, Uber and Upwork.

Horizontal EM makers facilitate the purchase and sale of goods and services used by a range of industries (Grieger, 2003). Horizontal marketplaces are also known as “functional marketplaces” because they cut across industries to optimise specific functions in an enterprise, including human resources, procurement, logistics and marketing (Korper & Ellis, 2000). Amazon, eBay and Ali express are good examples of horizontal EMs.

2.2.3. Benefits of Electronic Marketplaces

With the rise of online marketplaces, businesses that want to go online or implement e-commerce in their traditional commerce do not have to worry about the barriers previously discussed.

The owner of the marketplace handles all the issues that come with e-commerce which allows the users (both buyers and sellers) to trade swiftly. Electronic marketplaces benefits to the users are that (i) it provides easy and fast entrance to new markets (Kamel, 2006), (ii) increase in revenues (Khan, 2016) and increase of sales (Kamel, 2006) for the sellers, (iii) the negotiation cycles are lowered (Kamel, 2006) by omitting the ‘agents’, (iv) buyers have the ability to order at any time (Vidyapeeth, 2010) and sellers and business to operate full-time (Kamel, 2006), (v) lowering of unnecessary inventory (Khan, 2016), (vi) it promotes efficient production planning and development, (vii) not having the need for a physical store saves the costs that comes with owning a store such as store rent and transportation cost (Khan, 2016). EM also allows easy and fast

comparison between offers of many sellers, fast selection of best offers and provides order status tracking.

2.3. Electronic Marketplace Building Approaches

An online marketplace in its core, is an information system. All approaches to information systems development proceed through four stages: planning, analysis, design, and implementation. These four stages of systems development all fit into the development phase of the information systems life cycle (McKeown, 2009).

That being said, there are three approaches to building an online marketplace platform namely, internal development, outsourcing and acquisition (McKeown, 2009). The approaches will be discussed in detail below.

2.3.1. Custom development or ad hoc programming

This approach is feasible if the EM builder is a developer or has the budget to hire a developer, it allows the party build exactly the kind of platform they have in mind.

The benefits that come with this approach is that: the EM builder has access to all the source code, they are not dependent on third-party software, there are no licensing issues and no subscription costs.

Ad hoc programming has its downsides such as a considerable time consumption. (i) Building a multi-sided platform with a flawless transaction flow from scratch takes time. Other downsides are: (ii) A usable information system that often yields an information system that is poorly documented, poorly planned, not integrated into the overall organization information plan, and difficult to maintain and modify (McKeown, 2009). (iii) The need to maintain the site regularly, which is a lot of work. Technology, user experience trends, and even online payment regulation evolve quickly and (iv)

Continuous development work is required to improve the platform based on feedback, pay monthly hosting bills, and update and maintain the platform.

2.3.2. Off-the-shelf solutions

There are four off-the-shelf solutions namely Software as a Service (SaaS), Content Management System (CMS), Application Programming Interface (API) and Platform as a Service (PaaS).

2.3.2.1. Software as a Service (SaaS) solution

SaaS is a software model where software is accessed online via a subscription, rather than bought and installed on individual computers and is centrally hosted. Clients do not purchase software or infrastructure (e.g., hardware and OS) upfront, but pay for their access to the services over time (Goyal, 2013).

The SaaS solution implementation cycle is shortened, since applications are already deployed on SaaS vendors' sites (Goyal, 2013). It also allows extensive cost savings in operating standard business components on a large scale. Marketplaces with a simple business idea and without a technical team are advised to use the SaaS marketplace platform. The service will create an account, connect a domain and allow for a personalised payment method.

Also, firms with high cost of capital may find the SaaS model more beneficial as it enables them to economize on fixed capital cost by spreading the service cost over time, allows faster time to value, and potentially brings significant cost savings (Sharpe et al,1995)

SaaS effectively redefines the software deployment model from packaged applications with upfront licensing fees and lengthy implementations to one that constitutes a dynamic, “pay-as-you-go” Internet delivered service relationship (Goyal, 2013). One of the most common pricing models for such services is a monthly and yearly

subscription fee. The exact price may depend on the number of transactions made through the marketplace, the number of listings (published goods or services), or the total number of users that have been registered on the platform.

The most popular examples of SaaS builders for marketplaces are Sharetribe Go, Arcadier and MarketplaceKit. The hosted versions of these platforms are the same except for a few differing features. All three platforms offer hosted versions, but only two of these SaaS builders offer self-hosted versions, namely Sharetribe and MarketplaceKit.

The following are the advantages and disadvantages of Hosted and Open-source versions of these builders.

Hosted SaaS: (i) Take full responsibility for the technical part: installation, maintenance, updates, security, backups, (ii) Automatically updates to the latest versions, (iii) offer full technical support, (iv) Stripe and PayPal payments are available, (v) it is easy to shift from the hosted to the self-hosted version, (vi) the software is hosted on the vendor server, (vii) unable to customize the backend and (viii) there is limited customization of CSS, HTML, and JavaScript.

Self-hosted version/ Open source version: (i) allows the ability to rewrite the code, (ii) the software is installed on the builder's server, (iii) the builder handles each of the technical elements, (iv) the level of support depends on the voluntary help of the user's community, (v) some code and used methods are deprecated, (vi) it is price-free, but a customer needs to pay for their initial setup, customization, hosting and maintenance and (vii) strong technical knowledge is required.

2.3.2.2. Application Programming Interface

API is a set of requests and methods between a server and a client. Marketplace API or "API as a service" seems to be the natural evolution of SaaS marketplace technology.

All the API service providers want to offer **both** zero-cost management and maximum customization flexibility to their customers – people who want to build a

marketplace. A great solution can be building an API on top of their own backend, deployment, and monitoring systems and offer it as a service. However, the UI and specific marketplace builder's logic development is a burden for the marketplace development team.

Creating an SDK can make the life of a marketplace developer easier: a library (or set of libraries for different programming languages) wraps low-level calls to the API from marketplace front-end over HTTP or GraphQL and speeds up the development.

Moreover, some marketplace API providers offer a frontend boilerplate that uses their marketplace API (+ SDK). A particular marketplace developer may use it as a starting point to test the hypothesis and customize or even rewrite it from scratch later if needed.

API gives (i) the maximum level of the UI and business logic flexibility, (ii) the ability to integrate the API into existing e-commerce or marketplace websites, and (iii) it simplifies work for a marketplace builder in terms of API support, QA and DevOps.

The downsides of using API as the solution are (i) UI and UX for a new marketplace need to be created from scratch, (ii) hosting, deployment, and backups of the front-end part is still the responsibility of the marketplace builder and (iii) the flexible functionality is still limited. **Good examples:** Stelace, Sharetribe Flex and Arcadier, Izberg.

2.3.2.3. Platform as a Service

Goyal, 2013 defines PaaS as “*a category of cloud computing that provides a platform and environment to allow developers build applications and services over the internet*”. It is a blend of open-source CMS, API as a Service and vendor hosting, the UI and logic are provided by the platform (similar to CMS) and around its APIs, the deployment and administration are done through the platform portal interface.

The services offered by PaaS are *operating system, server-side scripting environment, database management system, server software, support, storage, network access, hosting, tools for design and development* (Goyal, 2013). PaaS provides the architecture as well as the overall infrastructure to support application development.

The advantage here is that it gives a great deal of much flexibility within the realm of customization. Users don't need to invest in physical infrastructure such as servers or employ the expertise to manage it (Goyal, 2013). Clients can have control over the tools that are installed within their platforms and can create a platform that suits their specific requirements. Also, the EM builder does not have to choose between front end and back end.

The downside is that the EM builder is tied to the platform's grid of deployment and scaling servers. This takes away the hassle with DevOps and support but causes vendor dependency in terms of deployment and scaling

2.3.2.4. Content Management System

A content management system (CMS) is a computer application that allows publishing, editing, modifying, and organizing, deleting, and maintaining content from a central interface (Srivastav & Nath, 2016). It provides the necessary infrastructure for multiple persons to effectively contribute content and collaborate throughout these lifecycles (Hill, 2010).

Marketplace CMS gives much more flexibility than custom SaaS in terms of logic and UI customization. That is because the developers in the marketplace team have full control over the code base and can create pages, animations, and transaction flows within almost any level of complexity.

There are two types of CMS: (i) Vendor-hosted: the vendor offers a customizable code base with additional hosting and deployment services, as well as backups and updates for the marketplace and (ii) Self-hosted: which can be an open-source marketplace CMS or a commercial lifetime license which provides the EM owner with

an instance of a closed-source CMS to deploy, scale and customize to any extent they need (without support).

CMS (i) provides good opportunities for UI and business logic customization and (ii) a marketplace builder can still rely on the vendor in terms of hosting, deployment, backups etc.

The disadvantage of using CMS is that the solution may become an additional development cost or even impossible if it doesn't feature public API or integration with external APIs

Good examples: WordPress, Sharetribe Go, Shopify, Magento, MarketplaceKit, Cocorico and BigCommerce.

Since there is no single solution that is right for every job (Messenlehner & Coleman , 2019), deciding which solution to use depends on several factors such as the marketplace builder's skills, financial ability, time available and what a particular solution has to offer. Putting all the factors into account, open-source content management software WordPress is selected for building a marketplace in the context of this paper.

2.4.WordPress as the Electronic Marketplace building solution

As the definition of an electronic marketplace states, an EM is a web application. In order to build an EM, it is important to understand the concept of a web application, its features and architecture. This section discusses the concept in detail.

2.4.1. Web application

2.4.1.1. Understanding a web application

A web application (app) may be defined as a software designed to perform a group of co-ordinated functions, tasks or activities for the benefit of the user which runs online through a web browser (Messenlehner & Coleman , 2019). It is a collection of files that

reside on a computer (web server) connected to the internet (Bodo, 2003). The following features best describe web apps:

They contain interactive elements such as forms, toolbars, drag-and-drop elements, rich text editors and sliders allow the web apps to have more interactive user interface.

Another feature is web apps allow its users to perform specific tasks. A good example is Google maps allowing its users to get driving directions.

Web apps save information of its users through several forms such as login forms and address forms. The information is used to facilitate tasks and enable persistent experience.

Applications are able to take advantage of the user's device capabilities such as the camera, microphone and the gps. They can also adjust depending on the device that is accessing it.

Web applications also have the capability to work offline. A good example is with Evertone, a user is able to draft notes offline and then synchronise them to the internet when connectivity is restored.

Mashups is another feature whereas web apps are able to tie one or more web apps together. Apps are able to use various web services and APIs to push and pull data.

2.4.1.2. Web app architecture (3-tier client-server model)

Web applications use the 3-tier client-server model whereas the software architecture is made up of both the client and server (Zhang, 2013) , whereby the clients sends requests while the server responds to the requests sent. 3-tier architecture involves the client, Database server and Application server (Kambalyal, 2010) also known as the front-end layer, data layer and business logic layer.

The front-end layer is also known as the presentation layer. This is the layer that external users interact with and perform their requests and receive the response. The main

function of the interface is to translate tasks and results in a manner that is understood by the user.

Business logic layer accepts data from the presentation layer and passes it to the data layer. It has the following functions: (i) it acts as an interface between presentation layer and data access layer, (ii) all business logic – like validation of data, calculations, data insertion/modification are written under business logic layer, (iii) It makes communication faster and easier between the presentation and data layer, and (iv) it defines a proper workflow activity that is necessary to complete a task.

The data layer receives the data from the business layer and performs the necessary operation into the database.

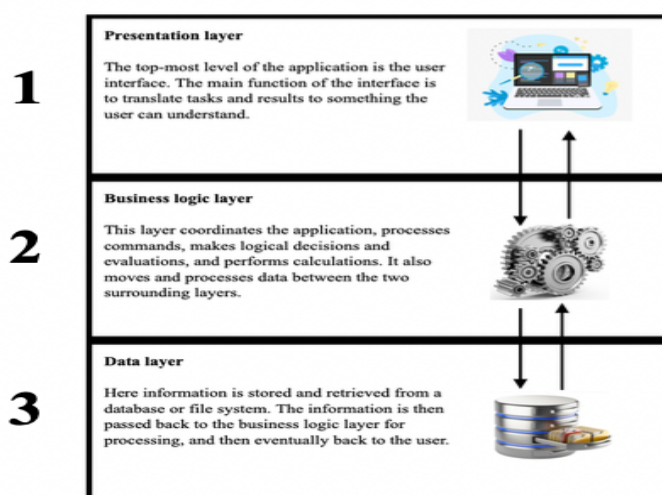


Figure 3 3-Tier Client-Server model

2.4.1.3. Benefits and uses of client-server model

This model has several advantages such as: Allowing easier sharing of resources from client to servers, reduces data replication by storing data on each server instead of client. and it splits the processing of application across multiple machines.

Examples of client-server applications are: File transfer protocol (FTP) for transmission of files such as movies and music between clients, Mail transfer protocol (MTP) which is used to transfer emails and Hypertext transfer protocol (HTTP) for transferring of multimedia files such as images, text between the client and server.

2.4.2. WordPress

2.4.2.1. Concept and features

Wikipedia best describes WordPress (WordPress.org) as “a free and open-source content management system (CMS) written in PHP and paired with a MySQL or MariaDB database”. It is a great framework (Messenlehner & Coleman , 2019) to building web applications.

WordPress features include a plugin architecture and a template system, referred to within WordPress as Themes. Other WordPress features are as follows:

- I. WordPress makes content management easy due to its WYSIWYG (what you see is what you get) editor tool. This allows the user to make quick and easy changes without using any code.
- II. User management such as adding administrative users and end users to the site is made easy and secure by the Roles and Capabilities system. This allows control and capabilities according to user levels. Plugs such as Paid Memberships Pro help in creating members of different levels and control the content the users can have access to.
- III. WordPress repository contains more than 55,000 free plugins ranging from e-commerce such as WooCommerce, forums plugins like bbPress, membership sites for example Paid Memberships Pro, social networks functionalities like the BuddyPress plugin and gamification plugin BadgeOS. There are also plugins that support SEO (search engine optimisation), google analytics and other common tasks

- IV. The technologies that are used to build WordPress applications (PHP, MySQL, JavaScript) are well suited for a wide range of tasks. Another technology which WordPress uses worth mentioning is Responsive Web Design.
- V. The developers behind WordPress perform frequent security updates which address the system's vulnerabilities. The architecture of the software makes applying of the updates quick and easy and by staying up to date by upgrading the latest version makes WordPress a "far more secure platform than anything else available" (Messenlehner & Coleman , 2019).
- VI. WordPress, it's plugins and the technologies used are all free. The only services that cost money are servers and hosting which vary depending on the site traffic. Another cost may be if there is a requirement for a custom functionality not found in the available plugins.

2.4.2.2. WordPress as a Framework vs Model-View-Frameworks

Model-View-Controller (MVC) is a design pattern used in many software development frameworks. MVC allows the developer to reuse the code by allowing models, views and controllers to interact. The model also supports separation of concerns (SoC) by allowing designers to focus their attention on the views while the programmers focus their attention on the models. WordPress does not use MVC approach but has similar development processes. WordPress also encourage reuse of codes and SoC.

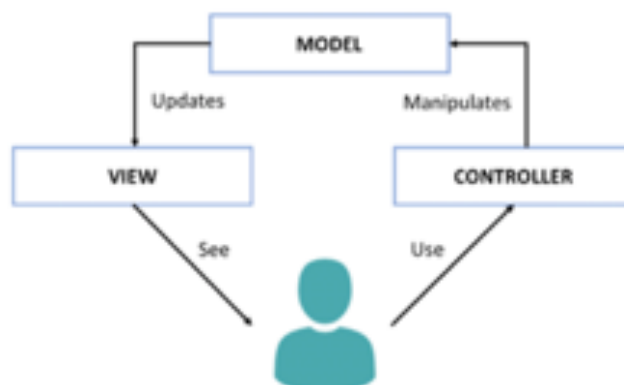


Figure 4 Model-View-Controller (MVC) Model

The figure above describes a typical MVC-based application. The end user uses a controller, which manipulates the application state and data via a model, which then updates a view that is shown to the user. The table below compares the two frameworks in action:

Table 3 Model-View Framework vs. WordPress (Messenlehner & Coleman , 2019)

MVC framework	WordPress
<p><u>Models</u> The code that stores the underlying data structures and business logic is found in the models. Back-end programmers are responsible</p>	<p><u>Plugins</u> New data structures, complex business logic and custom post type definitions are stored in plugins. Plugins also have functionality that is “views” like</p>
<p><u>Views</u> The code to display data to the users is written in the views. Front-end developers are responsible</p>	<p><u>Themes</u> The templating code and logic is stored in themes</p>
<p><u>Controllers</u></p>	<p><u>Template loader</u></p>

The code to process user input and decide which models and views to use to handle a request is stored in controllers Back-end programmers are responsible	All page requests are processed through the index.php file and processed according to template hierarchy
--	--

2.4.2.3. WordPress technologies (Themes, Plugins and Databases)

2.4.2.3.1. Presentation layer (Themes)

WordPress has a collection of over 7000 different themes in its *theme directory*. Themes determine the design of the website including its layouts (WordPress Org, 2020). It is the front-end of the site, what a visitor sees when they browse the site on the web. Themes take the content and data stored by WordPress and display it in the browser.

WordPress theme contains index.php as its main template file and style.css as the main style file. Other files are PHP files, CSS files, Graphics, JavaScript and Text files which contain the license info, readme.txt instructions and a changelog file.

WordPress takes advantage of responsive web development by using HTML5 and CSS3. W3Schools defines responsive web design as “*using HTML and CSS to automatically resize, hide, shrink, or enlarge, a website, to make it look good on all devices (desktops, tablets, and phones)*”.

Responsive design adapts the new model of interaction more complete, dynamic and personalized (courtesy of Web 4.0) in which there is a symbiotic interaction between humans and machines (Choudhury, 2014). It allows any website to dynamically adapt to the width of the device in which is being visualized.



Figure 5 Responsive Web Design Example

Responsive web design is composed by three technical components (Cazañas & Parra, 2017) : (i) fluid grids, (ii) flexible images; and (iii) media queries. Fluid grids are responsible to assign relative units to page elements; flexible images are also sized in relative units; and media queries make possible to switch between different CSS based on the features of the device.

There are several advantages associated with responsive design approach. Lestari et al. (2014) advocates that responsive web design is a suitable approach to offer improved user experiences. Majid et al. (2015) considers that responsive design is a highly flexible approach that can be used for the development of e-commerce solutions to improve SEO.

Rashid (2017) consider that there are five essential reasons that support the great growth of responsive design, respectively: (i) increased traffic from mobile users; (ii) lower cost and website maintenance; (iii) provides a seamless user experience; (iv) adapts easily to any screen size; and (v) improves SEO efforts.

Winkless (2015) suggests that responsive web design contributes to increase the productivity of web designers. Finally, Baker (2014) advocates that web responsive

design, namely the use of HTML5 and CSS3, promotes the creation of accessible web pages.

2.4.2.3.2. Business logic layer (Plugins)

PHP is a server-side scripting language that is embedded in HTML. It is used to manage dynamic content, databases, session tracking, and has the capability to build an entire e-commerce site. PHP language is known for its “*simplicity, efficiency, security, flexibility, familiarity and its server side*” (Ogala & Sylvester , 2019) functions.

WordPress uses PHP language as its business logic (WordPress plugins) programming language. WordPress codex defines a plugin as “*a program, or a set of one or more functions, written in the PHP scripting language, that adds a specific set of features or services to the WordPress weblog, which can be seamlessly integrated with the weblog using access points and methods provided by the WordPress Plugin Application Program Interface (API).*”

The following are the common functions of PHP language: (i) PHP performs system functions such as create, open, read, write and close, (ii) it can handle forms by gathering data from files, save data to a file, through email you can send data and return data to the user, (iii) user accessibility restrictions, (iv) access cookies variables and set cookies, (v) data encryption and (vi) database manipulation such as add, delete and modify.

WordPress plugin repository contains more than fifty-five thousand (55,000) plugins (WordPress Org, Plugins, 2020), some free and some premium which range from e-commerce to social network functionalities to gamification to multi-vendor marketplace functionalities plugins.

The following table shows some of the most important and best WordPress plugins:

Table 4 Best WordPress plugins and their functions.

PLUGIN	FUNCTION
MonsterInsights Lite	Google Analytics Plugin
WPForms Lite	Drag & Drop Form Builder
SeedProd	Coming Soon & Maintenance Mode Plugin
TrustPulse	FOMO & Social Proof App
WP Mail SMTP	Easy Transaction Email Tool
OptinMonster	Lead Generation Plugin
RafflePress	Amazingly Easy Giveaway Plugin
MemberPress	Membership Plugin
Smash Balloon	Social Feed Plugin
Beaver Builder	Page Builder
LiveChat	Conversational Marketing Chat Tool
Visualizer	Tables and Charts WordPress Plugin
YoastSEO	Leading WordPress SEO Plugin
WooCommerce	Top Plugin for eCommerce Sites
W3 Total Cache	Caching Plugin
Pretty Links	Must-Have Affiliate Marketing Tool
LearnDash	Best LMS Plugin
Smush	Top WordPress Image Optimization Plugin
Better Click to Tweet	Great Twitter Tool
Instagram Feed	Top WordPress Instagram Plugin
Envira Gallery	Awesome Image Gallery
Soliloquy	Best Slider Plugin
Optimole	Best Real-Time Image Optimization Plugin
PrintFriendly	Fantastic PDF, Email & Print Tool
Sucuri	Top Security Tool

According to WordPress codex, the main difference between WordPress themes and plugins is that *“a theme controls the presentation of content; whereas a plugin is used to control the behaviour and features of the WordPress site”*

2.4.2.3.3. Data layer (WordPress Databases)

A database is a structured, organized set of data. It refers to a software used to store and organize data. The main function of a database software is to respond quickly to requests, manage and save data (Stroe, 2011) in an efficient manner, meaning that the interrogation should be done in an acceptable timeframe. There are several database servers in the market such as Oracle, MySQL Oracle, SQL Server, PostgreSQL or IBM DB2.

WordPress runs on top of a MySQL database and creates its own tables (Messenlehner & Coleman , 2019) to store themes and plugins data and all contents. MySQL is a software used to create databases, store and get data when requested (Oracle). MySQL is also an open source software, just like WordPress and works best with other popular open source software, such as Apache web server, PHP, and Linux operating system.

When WordPress is first installed before adding any additional plugin, it installs as a blog and automatically installs 12 database tables as shown on the database structure diagram figure 2.7 and described on table 2.5.

In order to install WordPress, one must have a MySQL database which is usually offered by (web) hosting providers, included in their hosting packages.

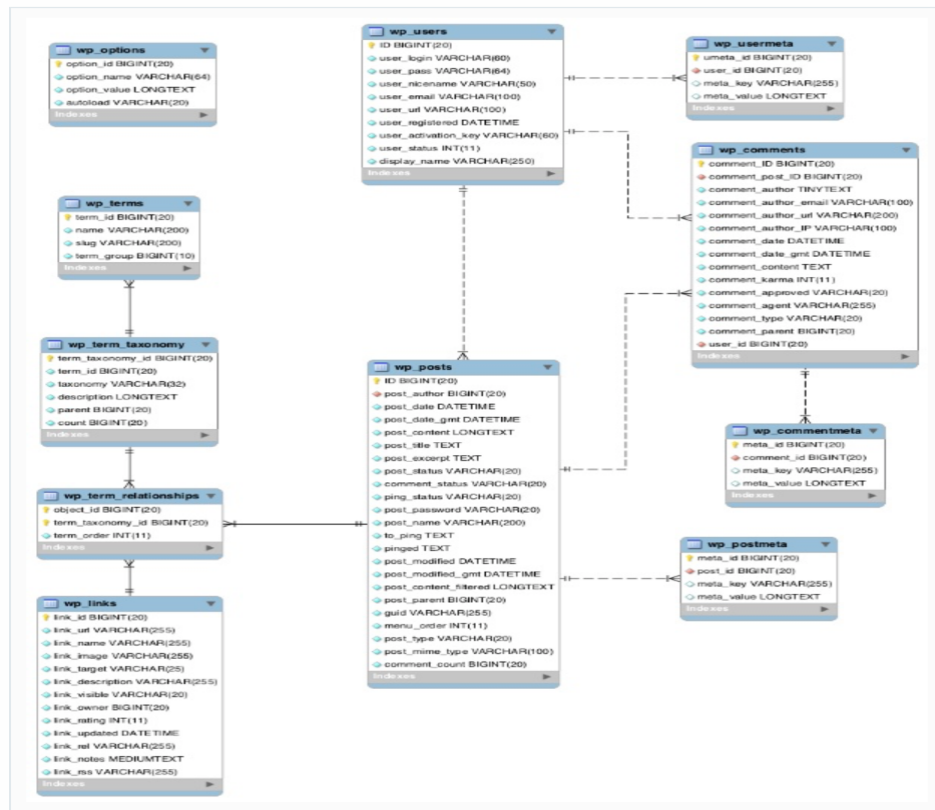


Figure 6 Custom WordPress blog database structure

The custom tables are described on the table 2.5 below

Table 5 Custom WordPress blog database tables overview

Database table name	Description
wp_commentmeta	This table contains meta information about comments posted on a WordPress website. The table has four fields meta_id, comment_id, meta_key, and meta_value. Each meta_id is related to a comment_id. One example of comment meta information stored is the status of comment (approved, pending, trash, etc.).
wp_comments	As the name suggests this table contains WordPress comments. It contains comment author name, URL, email, comment, etc.

wp_links	To manage blogrolls, create by earlier versions of WordPress or the Link Manager plugin.
wp_options	This table contains most of WordPress site-wide settings such as: site URL, admin email, default category, posts per page, time format, and much more. The options table is also used by numerous WordPress plugins to store plugin settings.
wp_postmeta	This table contains meta information about WordPress posts, pages, and custom post types. Example of post meta information would be which template to use to display a page, custom fields , etc. Some plugins would also use this table to store plugin data such as WordPress SEO information.
wp_posts	The name says posts but actually this table contains all post types, or should we say content types. This table contains all your posts, pages, revisions, and custom post types .
wp_termmeta	This table allows developers to store custom metadata for terms under their custom taxonomies. For example, WooCommerce uses it to store metadata for product attributes and categories.
wp_terms	WordPress has a powerful taxonomy system that allows a user to organize their content. Individual taxonomy items are called terms, and they are stored in this table. For example, WordPress categories and tags are taxonomies, and each category / tag inside them is a term.
wp_term_relationships	This table manages relationship of WordPress post types with terms in wp_terms table. For example, this is the table that helps WordPress determine post X is in Y category.
wp_term_taxonomy	This table defines taxonomies for terms defined in wp_terms table. For example, if you have a term “ WordPress Tutorials “, then this table contains the data that says it is associated with a taxonomy called category. In short, this table has the data that helps WordPress differentiate between which term is a category, which is a tag, etc.
wp_usermeta	Contains meta information about registered users on the website.
wp_users	Contains user information like username, password, user email, etc.

As plugins are added, more tables are created and also some tables functions are altered depending on the added plugins.

WordPress databases are managed using phpMyAdmin.

2.4.3. WordPress E-commerce

There are many different plugins that allow to accept payments on a WordPress site (Schäferhoff, 2018). Each has its strengths and weaknesses. Messenlehner & Coleman (2019) state the e-commerce plugins features as: (i) Integration with multiple payment gateways, (ii) Secure checkout forms, (iii) Saved order information and (iv) Products (or membership levels) with pricing.

WordPress ecommerce plugins generally fall into two main categories which are shopping cart plugins and membership plugins.

2.4.3.1. Shopping cart plugins

Shopping cart plugins are used for building web applications which offer products for sale (Messenlehner & Coleman, 2019). These plugins offer several features namely; a product custom post type, the ability to browse products, the ability to search through products, the ability to purchase multiple products at once, support for shipping addresses and shipping price calculations and support for custom tax rules (Messenlehner & Coleman, 2019).

WooCommerce is the most popular WordPress shopping cart plugin (Schäferhoff, 2018). It has three million active installs and powers 41% of all online shops which makes it the most popular ecommerce solution.

Shopify, Magento and BigCommerce are other popular solutions not powered by WordPress but offer an easy integration with WordPress site (Schäferhoff, 2018) which means that it is possible to connect a WordPress site with a Shopify or Magento or BigCommerce store.

Shopify launched Shopify for WordPress plugin in 2017 which allows users to sign up for their websites and create a store (Schäferhoff, 2018). Shopify then hosts the created shop for a monthly fee and handle the site maintenance and security. Shopify has the market share of about 6% with more than 275,000 vendors on the site.

Magento is another open source platform that can be installed on the user server (Schäferhoff, 2018). It is specifically aimed at large-scale projects and is also the most technical of the four solutions. Magento has a 6% ecommerce market share and is used by 11% of the top 10,000 popular websites.

BigCommerce is another ecommerce solution, similar to Shopify but with more features (Schäferhoff, 2018). All the user needs to do is sign up and design their store and then BigCommerce will host and maintain the site. The platform has over 95,000 live sites and with a market share of around 4%.

The table below explores the four ecommerce solutions which can be used with WordPress by their strengths, weaknesses and what purposes they are most suitable for.

Table 6 WordPress E-commerce solutions comparisons (Schäferhoff, 2018)

Ecommerce solution	Strengths	Weaknesses	Suitable use
WooCommerce	<ul style="list-style-type: none"> • Cost-effective • Easy to setup and use • Feature rich • Highly extendable • Customizable • Well supported • SEO and marketing • Full ownership 	<ul style="list-style-type: none"> • Advanced customization requires skills • Dependent of WordPress • Difficult to calculate true costs 	<ul style="list-style-type: none"> • Anyone who wants to take full ownership and control over their venture • Less experienced users • Scalable business
Shopify	<ul style="list-style-type: none"> • Ease of use • A wide range of features • Extendable • Mobile-friendly • Great support 	<ul style="list-style-type: none"> • Difficult advanced customization • Expensive 	<ul style="list-style-type: none"> • Beginners and people unwilling or unable to manage their own site
Magento	<ul style="list-style-type: none"> • Free (community edition) • Feature-rich (enterprise edition) • Third-party integrations • Customizable • Marketing-friendly • Scalable • Well supported 	<ul style="list-style-type: none"> • Unsuitable for beginners 	<ul style="list-style-type: none"> • Large-scale ecommerce websites • People with big budgets or coders

BigCommerce	<ul style="list-style-type: none"> • Easy to use • Feature-rich • Great design options • SEO and marketing • Wide range of payment options (60 gateways) • Customer support 	<ul style="list-style-type: none"> • Confusing plans • Expensive pricing • Lack of ownership 	<ul style="list-style-type: none"> • Small to midsize businesses • Anyone who wants to get started quickly
--------------------	---	---	--

Other shopping cart plugins include WP e-Commerce, Shopp Plugin, and Cart 66.

2.4.3.2. Membership plugins

Membership plugins are used for building web applications which focus around accepting payment for a membership access to the application (Messenlehner & Coleman , 2019). These plugins features include recurring pricing for subscriptions and tools for locking down content based on membership level.

There are several membership plugins namely Paid Memberships Pro, s2Members, Restrict Content Pro, Members, WPMU Membership and MemberPress. Of the named membership plugins, Paid Membership Pro is the only WordPress membership plugin that is 100% GPL and available for free in the WordPress repository (Messenlehner & Coleman , 2019). Other plugins have either paid modules or upgraded versions that are necessary to gain access to all of the plugin’s features.

Nearly every membership site has a slightly different way of calculating upgrades or special offers, or exactly how and when to lock down content. Instead of offering an extra-long settings page, Paid Memberships Pro carefully designed its hooks and filters to make it easy to setup nearly any pricing model with just a few lines of code.

Another key difference between Paid Memberships Pro and some other membership plugins is that Paid Memberships Pro uses its own table to define membership levels and their relationships to users and orders (Messenlehner & Coleman , 2019). Some membership plugins use the built-in WordPress user roles so

that each membership level is also a user role. User roles are very important in some membership sites, but in general, it's better to separate the concept of a membership level and a user role, allowing to have members who are admins *and* members who are subscribers. If you *do* need to assign roles based on membership level, that is easy to do with Paid Memberships Pro (Messenlehner & Coleman , 2019).

2.4.4. WordPress Electronic Marketplace

In order for a site to become a marketplace platform, different users (vendors) must have the ability to open stores and sell on the site. There are different plugins that can convert an ecommerce (WooCommerce) site into a marketplace, these plugins are known as multivendor plugins.

There are five best and most used multivendor plugins on the market namely Dokan, WC Vendors Pro, WC Marketplace, YITH Multivendor and WooCommerce Product Vendors. The chart below presents the statistics of the five multivendor plugins by number of active installations, number of features provided and plugin's price per year.

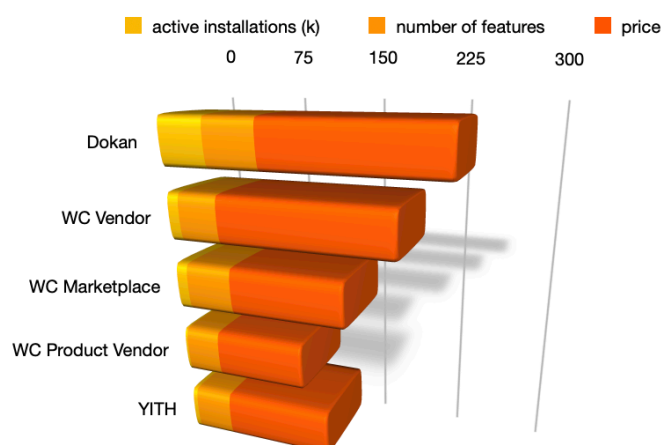


Figure 7 WooCommerce multivendor plugins stats

The table below presents the five WooCommerce-powered multivendor plugins descriptions, features and price per year.

Table 7 Multivendor Plugins features and prices

Plugin name	Features	Price
Dokan	Refund, Frontend dashboard for vendors, Frontend product management, Frontend order managing for vendors, Withdraw system for vendors, Individual vendor stores, Order email notifications, Seller management UI, Seller statement reports, Shipping management, Variable products, Coupon creation, Making announcements, Storefront widgets, Social login, Customized product categories, Store SEO, Social profiles (Facebook, Twitter etc.), Earning report, Back ordering system, Store specific contact form, Bookable product, Single product multiple seller, Seller verification, Trusted seller, Fixed commission for vendors, Multiple commission types, Product image, Limited time discount, Tax option, Front end order tracking module, General shipping options, Product level shipping, Product stock management, Update order status, Downloadable products, Product categories, Free version availability, Store header image, Custom store URL, Setup wizard, Store specific ticket system, Google maps, Manage reviews, Admin reports	From \$149 / year
WC Product Vendor	Refund, Frontend dashboard for vendors, Frontend product management, Frontend order managing for vendors, Withdraw system for vendors, Individual vendor stores, Order email notifications, Seller management UI, Shipping management, Variable products, Coupon creation, Storefront widgets, Customized product categories, Social profiles (Facebook, Twitter etc.), Earning report, Bookable product, Fixed commission for vendors, Multiple commission types, Product image, Limited time discount, Tax option, General shipping options, Product level shipping, Product stock management, Update order status, Downloadable products, Product categories, Free version availability, Store header image, Custom store URL, Admin reports	\$79 / year
	Refund, Frontend dashboard for vendors, Frontend product management, Frontend order managing for vendors, Withdraw	\$100 / year

<p>WC Marketplace</p>	<p>system for vendors, Individual vendor stores, Order email notifications, Seller management UI, Seller statement reports, Shipping management, Variable products, Coupon creation, Making announcements, Storefront widgets, Customized product categories, Store SEO, Social profiles (Facebook, Twitter etc.), Earning report, Back ordering system, Store specific contact form, Bookable product, Single product multiple seller, Seller verification, Fixed commission for vendors, Multiple commission types, Product image, Limited time discount, Tax option, General shipping options, Product level shipping, Product stock management, Update order status, Downloadable products, Product categories, Free version availability, Store header image, Custom store URL, Setup wizard, Google maps, Manage reviews, Admin reports</p>	
<p>WC Vendors</p>	<p>Refund, Frontend dashboard for vendors, Frontend product management, Frontend order managing for vendors, Withdraw system for vendors, Individual vendor stores, Order email notifications, Seller management UI, Shipping management, Variable products, Coupon creation, Making announcements, Storefront widgets, Customized product categories, Social profiles (Facebook, Twitter etc.), Earning report, Bookable product, Seller verification, Trusted seller, Fixed commission for vendors, Multiple commission types, Product image, Limited time discount, Tax option, Front end order tracking module, General shipping options, Product level shipping, Product stock management, Update order status, Downloadable products, Product categories, Free version availability, Store header image, Custom store URL, Admin reports</p>	<p>\$149 / year</p>
<p>YITH</p>	<p>Refund, Frontend dashboard for vendors, Frontend product management, Frontend order managing for vendors, Withdraw system for vendors, Individual vendor stores, Order email notifications, Seller management UI, Shipping management, Variable products, Coupon creation, Storefront widgets, Customized product categories, Store SEO, Social profiles (Facebook, Twitter etc.), Earning report, Bookable product, Fixed commission for vendors, Multiple commission types, Product image, Limited time discount, Tax option, General shipping options, Product level shipping, Product stock management, Update order status, Downloadable products, Product categories, Free version availability, Store header image, Custom store URL, Google maps, Manage reviews, Admin reports</p>	<p>\$95 / year</p>

Dokan has all the features out of all the plugins.

2.5. Web hosting

2.5.1. Web hosting fundamentals

In order for a web application to be accessible on the internet, it must have an address known as (i) a domain name which is the name that people type in the browser to open a certain website and a space known as (ii) a web server (Bodo, 2003). Web servers are costly and require technical expertise (Bodo, 2003) that most businesses and individuals do not have. This challenge brought about a solution, namely *web hosting*.

2.5.1.1. Web hosting definition

Web hosting is an act of providing storage space on the server, internet connectivity and services necessary to serve files and images (Puetz, 2005) to individuals or organisations, for the websites that are accessible via World Wide Web. Web hosts provide the place on the internet (Bodo, 2003) where the web application files, HTML codes, images and everything else is stored. The heavier the website, the more is the space you require to store its content.

Web hosts also provide bandwidth which refers to the amount of data that can be transferred to and from a server or a website. It is the allotted internet bandwidth that makes a website accessible to everyone online. The more the bandwidth, the better and faster is the network, connection and system. Bandwidth requirement is directly proportional to the number of visitors who visit a site. The more the number of visitors, the more is the bandwidth that's required.

The hosting provider also maintains the server, ensures website uptime and provides data security.

Hosting services are available as free or as a paid service. Free hosting is suitable for testing out a site or a service first (Stonewell, 2015). This is where it helps to try and

plan out the web application beforehand by calculating how much web space and bandwidth is required.

Paid hosting services vary depending on the features and packages of the particular hosting. Features such as reliable customer service, the control panel, templates, security, costs, the ability to upgrade and install multiple domains and the terms and services (Stonewell, 2015).

2.5.1.2. Types of Web Hosting

There are three types of paid web hosting servers, namely (i)dedicated servers, (ii)shared servers and (iii)virtual private servers.

Dedicated hosting is a type of internet hosting where a client leases a server which is not shared by anyone else (Stonewell, 2015). A single computer fully devoted to the needs of one customer (Bodo, 2003). They allow for faster access to information and provide the customer with greater flexibility to add advanced database and e-commerce applications. As the customer does not share RAM, disk space or a connection to the router, dedicated servers provide customers with the most advanced web hosting solutions.

Shared servers offer clients the ability to host their websites on a powerful, professional managed server, at a low monthly cost (Bodo, 2003). Shared servers provide individual websites with redundant connectivity, 99% uptime and 24/7 monitoring for a fraction of the cost of a dedicated server.

Virtual hosting is when numerous domain names are shared on the same server. The server will share certain features like RAM and CPU without needing these services to be pointed to the same hosting name (Stonewell, 2015). Virtual private (VPS) servers combine the flexibility of a dedicated server with the economical prices of shared server hosting (Bodo, 2003). VPS websites reside in private and protected areas on a shared

server. They maintain an independent set of applications, allowing the client to share the expenses of hardware and network connections while maintaining a secure allocation of RAM and CPU.

Table 8 Types of Web Hosting Servers (Bodo, 2003)

	Shared Server	Virtual Private Server	Dedicated Server
Intended clients and uses	Entry-level hosting needs	Entry-to-mid-level hosting needs	Mid-to-advanced-level hosting needs
Server response time	Activities and the number of other sites can affect server performance	Not affected by activities on other sites but allocated resources and bandwidth are limited	Depends solely on the volume of activity on the customer's website
Flexibility and scalability	Upgrade capacity is primarily focused on disk size and throughputs	Upgrade capacity depends on disk size, number of processors and throughputs	All aspects of the platform are upgradeable
Software	Limited to the software hosting provider installs	Many software options	Unlimited ability to download software programs to the server
Technical skills required	No technical skills required	Moderate	Advanced technical knowledge required
Cost	Cost-effective	Lower than dedicated server, higher than shared server	High

2.5.1.3. cPanel Service

Most hosting providers offer cPanel service. cPanel allows a user to manage and install websites easily. The user can upload files and documents through the file manager. cPanel also offers Awstats which track site visitors.

Stonewell. A (2015) defines cPanel as “*a UNIX based Web hosting control panel that provides a graphical interface and automation tools designed to simplify the process of hosting a web site. cPanel utilizes a 3-tier structure that provides functionality for administrators, resellers, and end-user website owners to control the various aspects of website and server administration through a standard web browser.*”

2.5.1.4. SSL Secure server

An SSL (Secure Socket Layer) secure server is a protocol which provides encrypted communication to a secure server (Stonewell, 2015). To use an SSL secure server, one must purchase an SSL certificate either through their domain name registrant or one of the different companies offering the service.

SSL certificates are useful if the web application requires personal and/or confidential information such as names, addresses and credit card information. SSL certificates encrypt data in order to prevent the interception and stealing of the data. They also prove the legitimacy and trustworthiness of the person or business owning the web application. This will in turn attract more visitors who are willing to spend money on the site.

2.6. WordPress Security

Messenlehner & Coleman , 2019 state several measures to take in order to protect a WordPress application from attacks. The measures are grouped into six actions as described below;

2.6.1. Basic security measures

These are the simplest but most important security measures to consider. The first basic measure is frequent updates. It is important to always upgrade to the most recent version of WordPress and the installed plugins and themes as soon as new versions become available.

Another key important measure is to change the admin default username “Admin”. When installing WordPress, the default username will be “admin” unless it is specifically changed. This is due to the fact that many bots will automatically try to login to a WordPress site with the username “admin” knowing that most people do not change the default username. All posts created by the admin account must also be changed to the new account.

The use of string password is another important measure to be taken into account. Choosing a secure password is very important especially for the administrators accounts. A good password is suggested to have a combination of upper and lowercase letters as well as numbers and special characters and above all it should be at least 10 characters long..

2.6.2. Hardening WordPress installs

These are the techniques that make it harder for the application to get hacked. The first technique is to disable the default functionality that allows administrators to edit the source code of any plugin or theme directly in the web browser. This is important so just in case a hacker is able to login to one of the admin accounts, he can't add any malicious code via the administrator. To disable the functionality, the code below must be added to the wp-config.php file:


```
<?php  
define('DISALLOW_FILE_EDIT', true);  
?>
```

The standard WordPress install uses wp_ as a prefix for all tables in the database. By changing this prefix, the site will become less vulnerable to SQL injection attacks. In order to perform this change, the following steps are to be followed:

1. Backup the current database
2. Open wp-config.php and change
\$table_prefix = 'wp_';
to
\$table_prefix = 'anyprefix';
3. Update the existing table names to include the new prefix with the following commands using phpMyAdmin (these few are examples)

```
rename table wp_commentmeta to anyprefix_commentmeta;  
rename table wp_comments to anyprefix_comments;  
rename table wp_term_taxonomy to anyprefix_term_taxonomy;  
rename table wp_usermeta to anyprefix_usermeta;  
rename table wp_users to anyprefix_users;
```

Using SQL commands, any instance of wp_ must be updated in the prefix options and anyprefix_usermeta tables and change any values like wp_ to prefix_:

```
update anyprefix_options set option_name = replace(  
option_name,'wp_','anyprefix_');  
update anyprefix_usermeta set meta_key = replace(  
meta_key,'wp_','anyprefix_');
```

4. Another option is using plugins to change the table prefix. The plugins are: *Change Table Prefix & Change DB Prefix*

Another technique is to move the wp-config.php file to one level above the WordPress install, which in most cases is the nonpublic directory. This file stores the

valuable information like the database location, username, password and the WordPress authentication keys. Since these values are stored in PHP variables and they are not displayed to the browser, it is not likely that anybody could gain access to this data, but it could happen.

Hiding login error messages if a user has entered the wrong username and password will prevent hackers from knowing exactly what they are doing wrong or right when attempting to access the site. This can be achieved by adding a line of code in the theme *functions.php* file or in a custom plugin which will hide or alter those messages:

```
<?php
add_filter( 'login_errors',
create_function(
'$a', '"Invalid username or password."'
) );
?>
```

Hiding the WordPress version is another security measure worth taken. A lot of bots scour the Internet in search of WordPress sites to target specifically by the version of WordPress they are running. They are looking for sites with known vulnerabilities they can exploit. By default, WordPress displays the following code within the `<head></head>` of every page:

```
<meta name="generator" content="WordPress 3.8.1" />
```

The version of WordPress can be hidden by implementing the following code:

```
<?php
add_filter( 'the_generator', '__return_null' );
?>
```

Preventing logins via `wp-login.php` by redirecting `wp-login.php` to the homepage is also an important security technique. This prevents bots from specifically trying to login via `wp-login.php`. Some bots are smarter than others. Hiding WordPress version from bots is important, but sometimes just knowing that you are using WordPress is all a bot may be looking for, and this is pretty easy if it sends a POST request to `wp-login.php`. Once a bot realizes `wp-login.php` exists, it can then begin to try to login to

the site. The following steps will make an alternative login page and hide the default *wp-login.php* login page:

1. Add the following rewrite rule to your *.htaccess* file:
RewriteRule ^new-login\$ wp-login.php
2. In your theme *functions.php* file or in a custom plugin, add this code:

```
<?php
function tznini_wp_login_filter( $url, $path, $orig_scheme ) {
    $old = array( "/(wp-login\.php)/" );
    $new = array( "new-login" );
    return preg_replace( $old, $new, $url, 1 );
}
add_filter( 'site_url', 'tznini_wp_login_filter', 10, 3 );

function tznini_wp_login_redirect() {
    if ( strpos( $_SERVER["REQUEST_URI"], 'new-login' ) === false ) {
        wp_redirect( site_url() );
        exit();
    }
}
add_action( 'login_init', 'tznini_wp_login_redirect' );
?>
```

The following plugins can also prevent logins via *wp-login.php*: *iTheme security & WP Admin*.

Adding custom *.htaccess* rules for locking down *wp-admin* is another important security technique for a WordPress app. For an application with only one or a handful of backend users, backend access can be restricted by certain IP addresses. This can be done by creating a new *.htaccess* file in the *wp-admin* directory of the WordPress install and add the following code, replacing 127.0.0.1 with the actual external IP address.

```
order deny, allow
allow from 127.0.0.1 #(repeat this line for multiple IP addresses)
deny from all
```

The same above code can be used to block IP addresses hitting the application that are suspicious either being bots or malicious users. However, the blocked users can get around by using a proxy server.

For backend users IP addresses that change often or there is way too many backend users, in order to manage all of the IP addresses, one can add a separate username and password to access the *wp-admin* directory. This adds a nice second layer of authentication because all of the backend users will be required to enter an htaccess username and password and their standard WordPress username and password:

```
AuthType Basic  
AuthName "restricted area"  
AuthUserFile /path/to/protected/dir/.htpasswd  
require valid-user
```

A *.htpasswd* file must be created somewhere in a directory above or outside of the WordPress install. In this file, a username and password must be added. The password can't just be plain text; a tool like *htaccess password generator* must be used to create an encrypted password. So the username/password for:

```
letmein/Pr3tTyPL3a$3!  
after encryption should be:  
letmein:E5Dj7cUaQVcN.
```

The entire encrypted string *letmein:E5Dj7cUaQVcN*. Must be added to the *.htpasswd* file; and when users try to go to */wp-admin*, they will be prompted for a username and password. The backend users must also know what this username and password is and must be kept private.

2.6.3. Backing up

It is important to make regularly scheduled backups for the site's content (database) and the *wp_content* folder. This makes it easier to restore the site in the event that it does fall victim to a hacker. It is recommended to schedule a backup at least once a week depending on the amount of new content being added. Daily backup is the best option.

Apart from manual backups, there are several plugins that can perform backups. The first backup plugin is *backup buddy*, this plugin works to make backups of all of the content on the WordPress site for safekeeping, restoring, or moving your site. Backups can be scheduled on a recurring basis and the file can then be downloaded to the computer, emailed, or sent off to the storage location of choice such as Dropbox or an FTP server. This plugin also features a restore option that will easily restore themes, widgets, and plugins. Backup Buddy also allows to easily move the site to a new server or domain right from the WordPress dashboard, which comes in handy if one works on a dev server and then move the sites over to a production environment upon launch.

VaultPress is another plugin created by the team at Automattic and offers users the opportunity to have all of their site content backed up in real time on cloud servers. Once installed, this plugin will automatically detect any changes to the content on the site as well as site settings and then update the backup copy with those changes. The plugin also features a one-click database restore in the event that the site ever becomes compromised. This is a premium plugin, meaning there is a fee for service, and different levels are offered. The premium version of the plugin also includes a daily security scan of the site to detect any issues as well as fixes for those issues.

2.6.4. Scanning

Scanning or monitoring the application is essential to know if there has been an attack. If the application ever gets hacked, it is important to know right away so that the issue can be addressed immediately. It is important to be proactive about protecting the web application against malware. There are several services that will scan the web applications such as Sucuri. Not only will Sucuri find malware and send an alert if the application has been infected, but it will also clean it up. Sucuri also has a great security plugin for WordPress.

Other scanner plugins are *WP Security Scan* which is a free plugin that performs a scan on the site and detects any areas of vulnerabilities in the site's security. It then suggests fixes for any of the issues it finds. One of the important security issues this plugin

helps with is changing the database table prefix. It also helps to hide which version of WordPress that is being used, which is information that hackers look for to use against when attacking the site. This plugin was developed by WebsiteDefender.com, which also offers a service to monitor the website for potential security threats, including malware and hacker activity.

Exploit scanner scans through all the files on the site and also sends an alert if it comes across anything that looks like it could be a potential threat. *BBQ* (Block Bad Queries) works as a type of firewall for the site by scanning all incoming traffic and then blocking all kinds of different malicious requests. Last but not least is *Antivirus-Once*. Like its name, once this plugin is installed and activated it runs a daily scan on the theme template files and database tables and sends an alert of any potential problems with email notifications. It will also add a message into the WordPress admin bar to alert of any viruses.

2.6.5. Security plugins

Apart from the above backup and scanning plugins discussed above, there are more very useful and powerful WordPress plugins that will help increase security for the application and also helps to recover quickly in case of a malicious attack.

Spam-blocking plugins such as *Askimet* and *Bad Behavior* are used to block comment spam from getting through to the site and works to block link spam from the site and functions best when run in conjunction with another spam service respectively. Bad behavior plugin looks at the content of the spam and the method through which the spam is being delivered by the spammer and the software being used, and blocks that all well.

On the other hand, *Askimet* plugin will run the site comments through a series of tests to ensure it is a real comment, and if it is identified as spam, it is automatically moved to the spam folder in the dashboard. This saves tons of time from having to sort

through all of the comments and determine which ones are spam or legitimate comments.

Login and password-protection plugins such as *Limit Login Attempts* and *Ask Apache Password Protect* are another class of security plugin. Limit Login Attempts is a great plugin to fight against brute-force attacks like someone running an automated script that will try to login to WordPress using random combinations of words. By default, WordPress will allow an unlimited amount of login attempts. This plugin limits the number of login attempts. If someone tries x times to log in and fails each time, she will be blocked from attempting to log in again for a set amount of time.

Ask Apache Password Protect is different from other WordPress security plugins in that it works at the network level to prevent attacks rather than at the site level. The user chooses a unique username and password that then protect login page and entire *wp-admin* folder. This plugin does require the use of an Apache web server and web host support for *.htaccess* files.

3. Architecture Framework

3.1. Domain Model

Figure 3.1 is the domain model of tznini marketplace platform. The marketplace platform has three kinds of users namely: (i) vendors posting their products for sale on the platform, (ii) buyers of the posted products and (iii) the platform administrators (admins) who moderate and keep track of the platform vendors, buyers and the posted products.

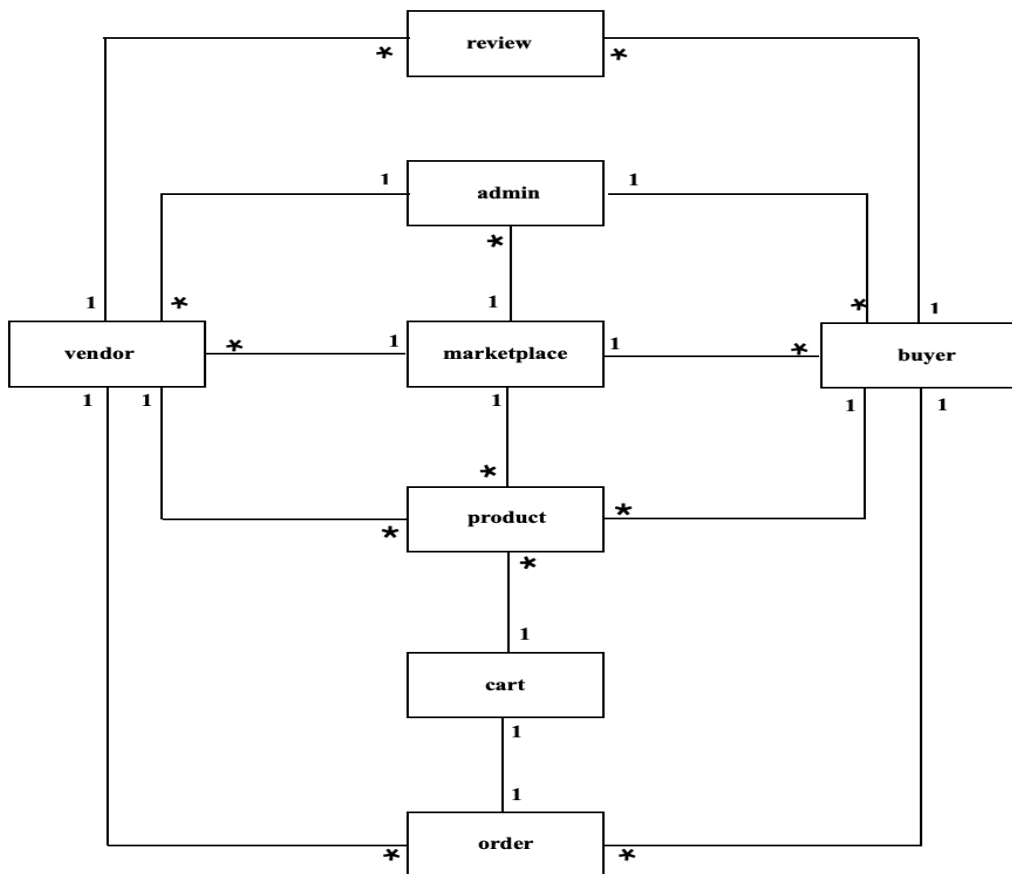


Figure 8 tznini domain model

Vendors and buyers can register on the marketplace but also the marketplace has more than one admin who have the ability to approve or disapprove, add and delete

vendors, buyers and products on the platform. One admin can approve/ disapprove, add or delete more than one vendor, buyer or product but a particular vendor, buyer or product can be approved / disapproved, added or deleted by only one particular admin.

The marketplace platform has multiple vendors. A vendor can post multiple products to sell on the marketplace platform and each particular product post can be owned by only one vendor. The marketplace also has multiple buyers whereas a buyer has the ability to view and purchase one or products, but the particular purchased product can only be bought by one buyer.

When a buyer decides to make a purchase, he adds the product / products in the cart. After the product / products added in the cart are paid for, their status change into an order. An order is then processed by the responsible vendor. One vendor can process more than one order at a given time, but an order can only be processed by one vendor.

A buyer also has the ability to write a review about a vendor and a purchased product. One buyer can review more than one product and vendor. A vendor can have many reviews, but one review can only be directed to one particular vendor.

3.2. Use Cases

Figure 3.2 illustrates the use case scenario of a vendor and an admin on tznini market. A vendor can register or be added by an admin in order to use the marketplace. If he is an existing user then must login into the system. An admin can delete a vendor for different reasons such as violating the platform rules. An approved vendor can customize his dashboard by adding a store banner, profile photo and update vendor information.

A vendor can post a product or products for sale on the marketplace platform. If the product does not meet the marketplace standards, the vendor can disapprove the product post or delete it. An approved product post stays up on the marketplace

platform until it is bought. If it is bought then the vendor processes the order by assigning shipping. The admin then updates the processed order status by adding shipping details on the order.

Once the order is completed by arriving to the intended party, the vendor can withdraw his earnings. The vendor can also update the product stock details. The vendor can logout from the marketplace platform.

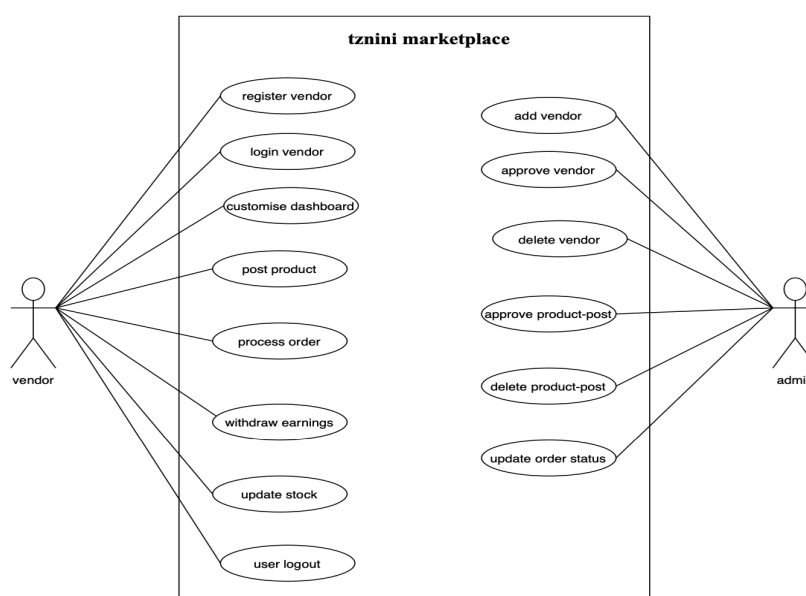


Figure 9 Vendor on tznini Marketplace platform

Figure 3.3 illustrates the use case scenario of a buyer and an admin on tznini market. A buyer can register or be added by an admin in order to use the marketplace, but he can also use the platform as a guest user (without registering). If he is an existing user then can login into the system. An admin can delete a buyer for different reasons such as violating the platform rules. An approved buyer can update his information such as address and billing information. A buyer can browse products and add desired product or products in the cart. The buyer can view the cart and edit it by adding or deleting products from the cart.

The buyer can confirm a purchase through checking-out from the cart by making a payment or place an order (if will pay cash upon arrival). Once the buyer receives his purchased product has the ability to write a review about the purchased product. An admin must review the buyer review post before it is posted on the platform. The buyer can logout from the marketplace platform.

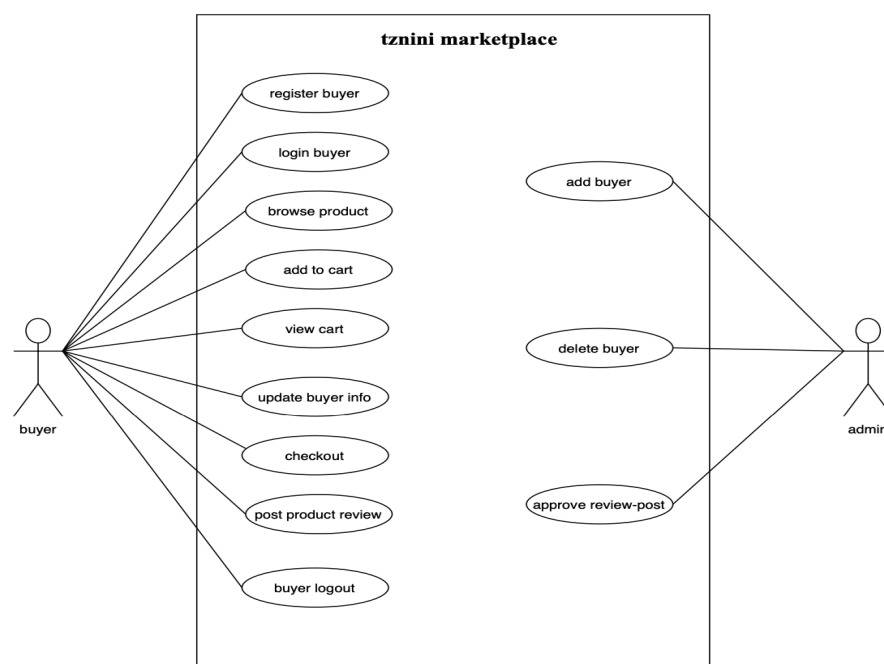


Figure 10 Buyer on tznini Marketplace platform

3.3.Database Structure

3.3.1. Tznini Database Structure Diagram

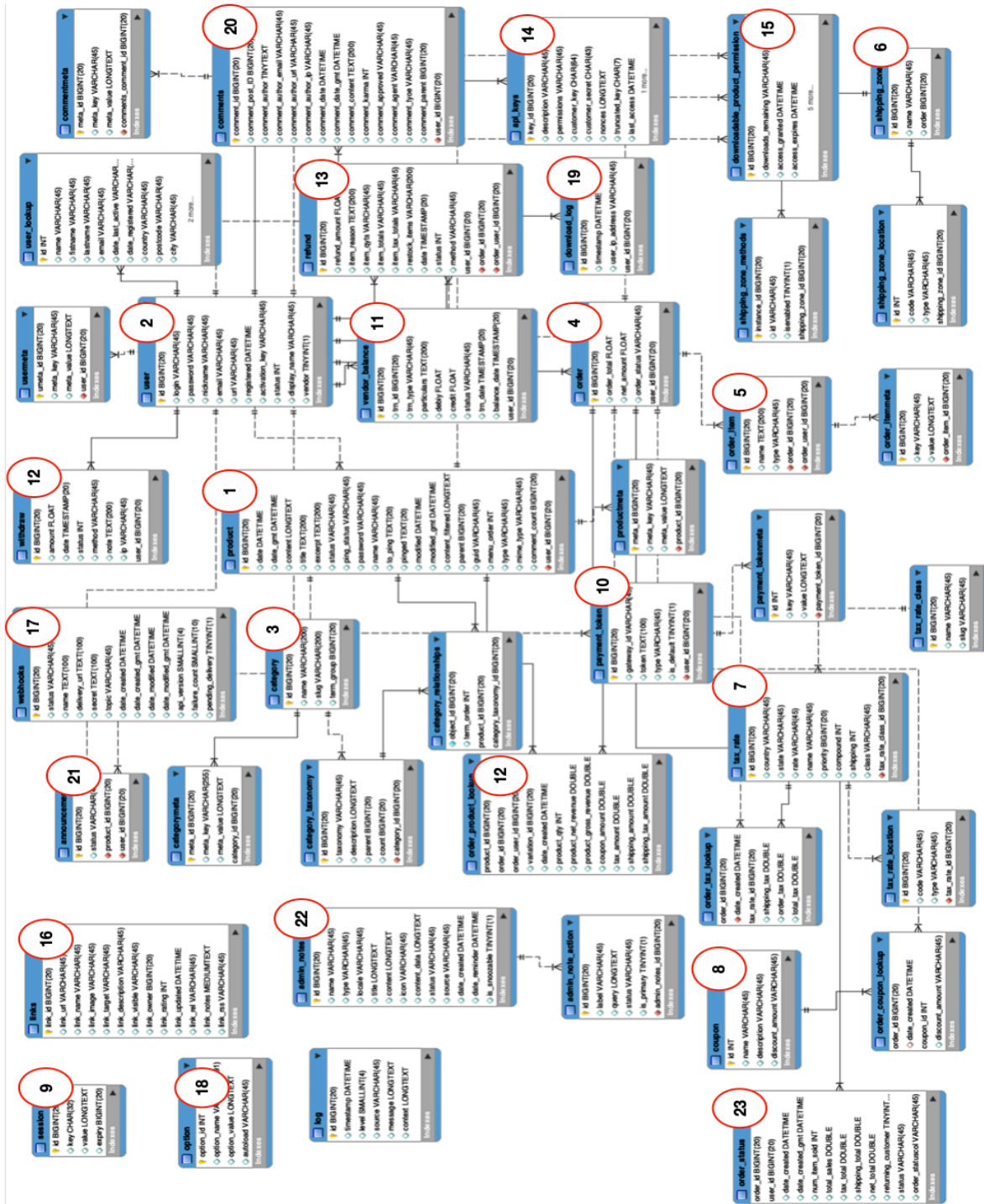


Figure 11 tznini database structure

3.3.2. Tznini Database Tables Overview

Table 3.1 contains a list of tznini database tables with tables description details.

Table 9 tznini database tables overview

No.	Table name	Description
1	product	The core of the platform data is stored here as well as navigation menu items are stored in this table
2	user	The list of users is maintained on this table
3	category	The categories of products and links and the product tags are stored
4	order	Stores order details
5	order_item	Stores line items which are associated with orders.
6	shipping_zone	Stores shipping zones you create in the settings area.
7	tax_rate	Stores tax rates defined in the admin area.
8	coupon	Stores details about the assigned coupons
9	session	Stores customer session data, such as carts.
10	payment_token	Stores customer payment tokens (used by gateways).
11	vendor_balance	Stores the amount of money a vendor has from sales
12	withdraw	Records the withdraws the vendor has done
13	refund	Records refunds details
14	api_keys	Stores API Keys used for the REST API
15	downloadable_product_permission	Stores product access permissions for downloadable products (granted after purchase).
16	Links	Links holds information related to the links entered into the links feature of WordPress
17	webhooks	Stores any webhooks that have been setup on the store.
18	options	Administration settings panel details are stored in this table
19	download_log	Logs user downloads of downloadable products.

20	comments	All comments within the platform are stored here
21	announcement	
22	admin_notes	
23	order_status	Records the status of the order

4. System Prototype

4.1. General description

TZNINI which is a combination of two words: TZ(Tanzania) and NINI (what thing in Swahili) is an electronic marketplace application aimed to be an ideal technology innovation implementation for all vendors that conduct their businesses at Kariakoo market to post their products online and for buyers to access the market’s products, vendors and information as well as make purchases

4.2. User Interface

4.2.1. Interface Structure

Figure 4.1 is the interface structure diagram of tznini marketplace platform.

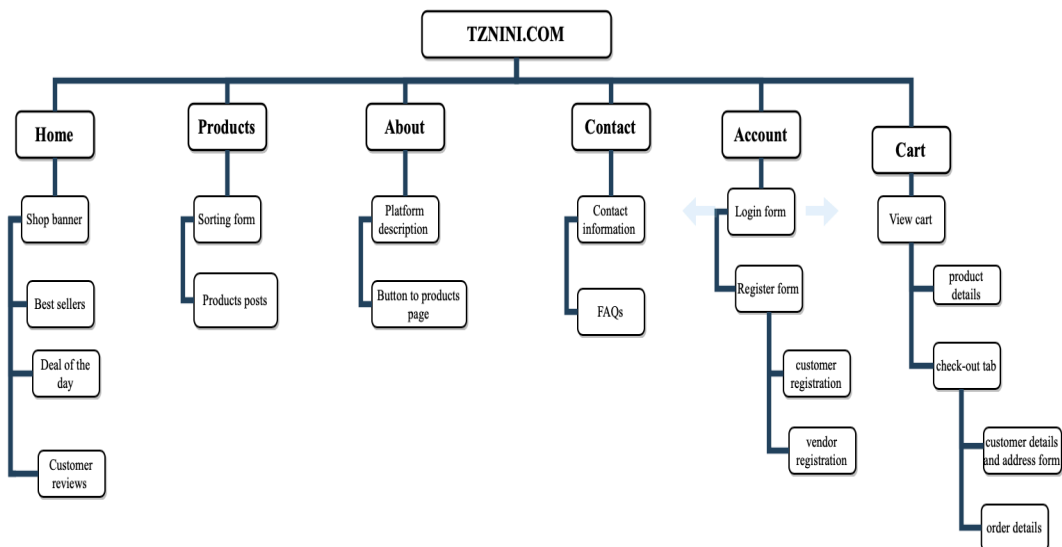


Figure 12 tznini interface structure

The marketplace homepage displays the list of options on its navigation menu.

Vendor must register or login to post a new product and manage their store. A buyer may sign up or login for a customised experience although they can buy on the platform without having an account.

Vendor adds a new product that he intends to sell on the platform. He has the ability to add the product details such as the product image, stock, description, category and more.

The new product is added on the products post page. The only products available at the marketplace are physical products.

User navigates to the products page. A list of products is displayed by images, each with the vendor details, its category, description and reviews. The products display can be arranged in the order the user wants. There is a sorting tab that allows a user to view/arrange product posts by date, price, relevance and type.

Buyer views a product and chooses whether to add to the cart or go back to shopping. The buyer can also write a review about the particular product.

Buyer adds the product to the cart and views the cart. In the cart, he has the option to add a coupon for a discount.

Buyer is checking out to confirm order purchase. The buyer has to pay upon order reception in cash.

For another scenario after the buyer adds a product in the cart and views the cart, the buyer chooses to change the product. buyer chooses to delete a product previously added into the cart, and to replace it with another product. The buyer can however increase or decrease the number of products in the cart. In case of other changes such as colour and size, the buyer must delete the current product and add the desired product.

As an alternative to completing the purchase in a single session, buyer decides to save the added products to continue later. This is only possible for the buyer who had logged in during the session.

Buyer logs off

Later buyer login, completes the order, checks out, and logs out. The user does not have to perform the buying process from the beginning. He can proceed from where he left off.

The buyer has now completed the order. The buyer selects an option that submits the order to the vendor. If the buyer has a special request about the order, has a section where he can write the special instructions before check-out. The vendor receives the special instructions along with the order request.

Buyer now wishes to change an order. The platform does not permit changes once the order has been submitted. The buyer receives the order confirmation and details through email and can view it in the platform in his account section.

The vendor receives the order made and start processing it by assigning the shipping method and ship the order. As the process goes, the order status changes, and the buyer can track the changes. The vendor may opt to send an email to the buyer about the order shipment details.

Later buyer logs in to check the order status.

Order is delivered.

The order may not be in the condition the buyer requested. He has the right to a refund or an exchange depending on the kind of product.

Buyer requests a return. The return or refund must be requested upon the arrival of the product and must be inspected by the courier that delivered it.

4.2.2. User Interface Screenshots

4.2.2.1. Homepage

Figure 4.2 shows the page that the user sees when he opens tznini web application URL. It is simple and maintains a clean uncluttered UI so that the user does not get overwhelmed after opening the application.

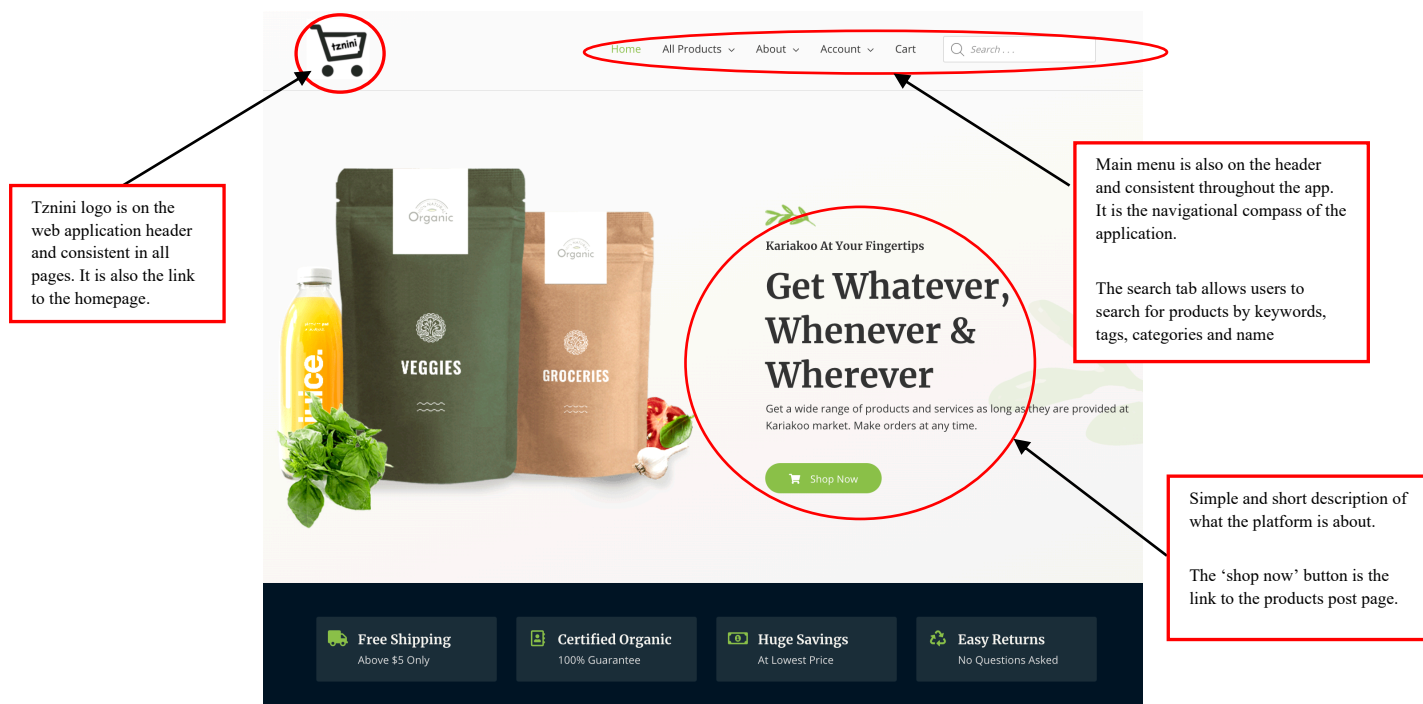


Figure 13 tznini homepage UI

4.2.2.2. Products post page

For a user that has opened the site for the first time, the product page (Figure 4.3) is most likely to be the next page he opens. It showcases all the products that are present for sell in the marketplace.

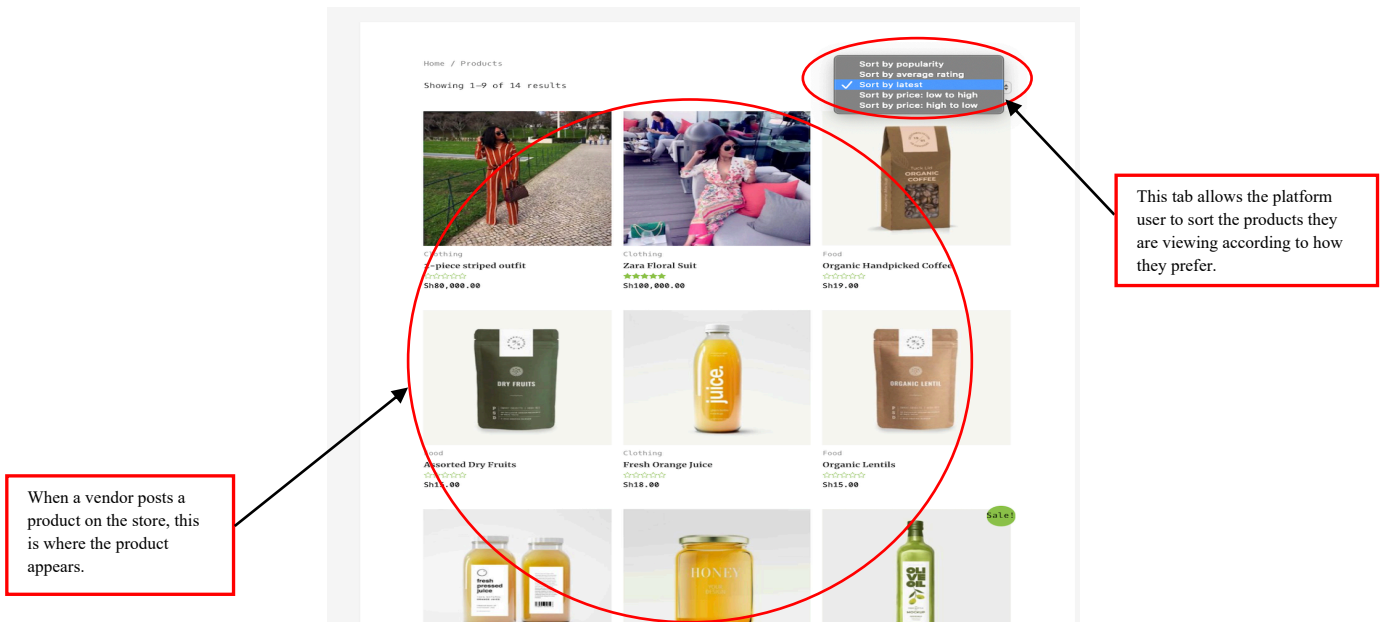


Figure 14 tznini products post page

4.2.2.3. Vendors page

Figure 15 is the vendors page which displays the list of all vendors in the platform.

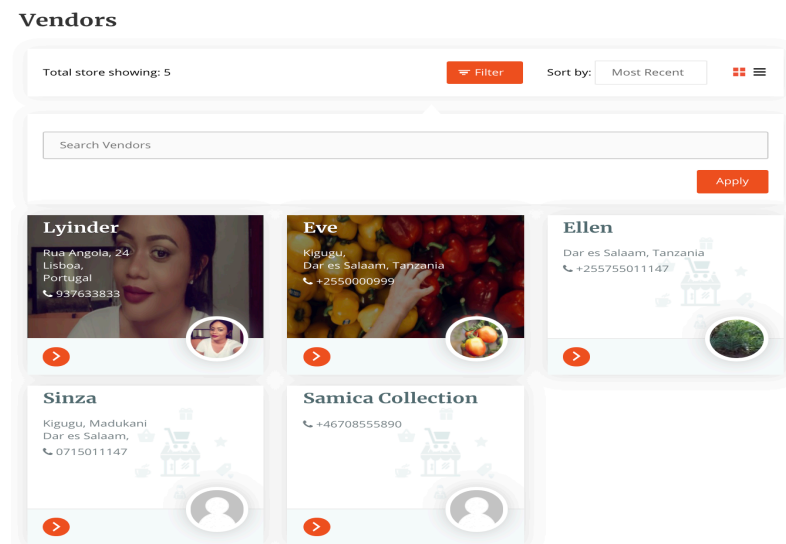


Figure 15 Vendors page

4.2.2.4. Information about the platform

The information page shown on Figure 4.4 gives a brief description about tznini platform as a business. This helps the platform user to be familiar with the company behind the platform.



Figure 16 tznini information about us page

4.2.2.5. Contacts page

Figure 4.5 shows the contacts page which displays all important information on how to reach the people in charge of the platform in case of any enquiry. It also features a 'Frequently Asked Questions' panel. It has the answers of the most common queries which may simplify the communication hassle for the user and save time.

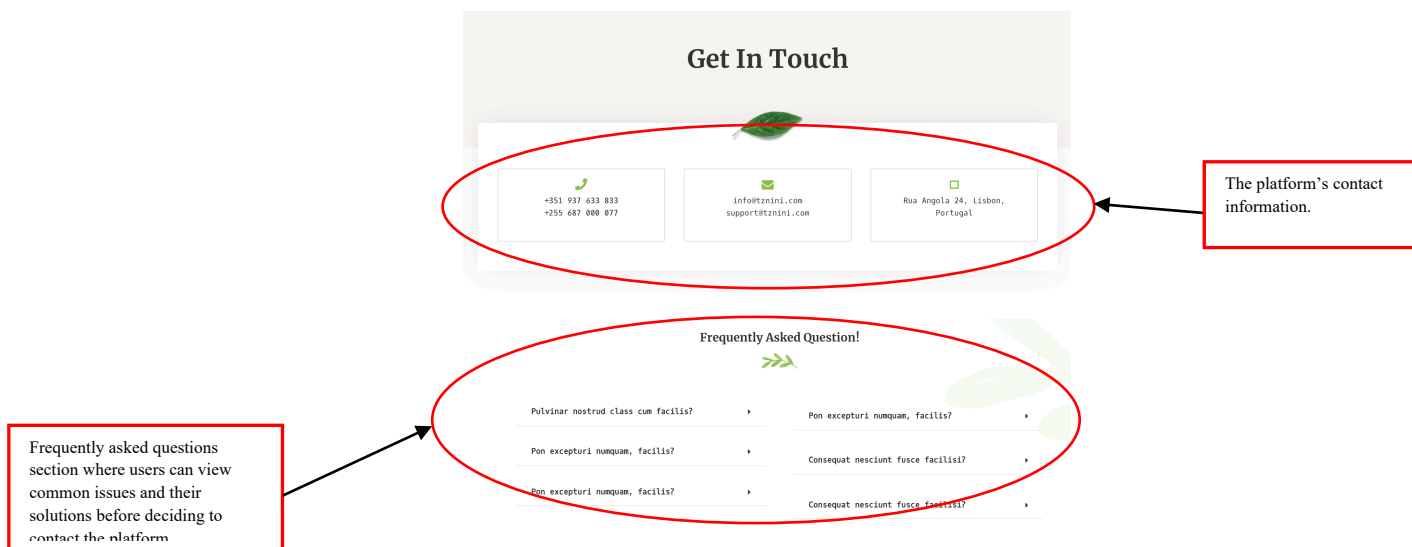


Figure 17 tznini contacts details page

4.2.2.6. My account page

For vendors to be able to sell their products, they must have an account on the platform. This is not necessary for a normal buyer although the option exists for them. This page allows users to register and login into their user accounts.

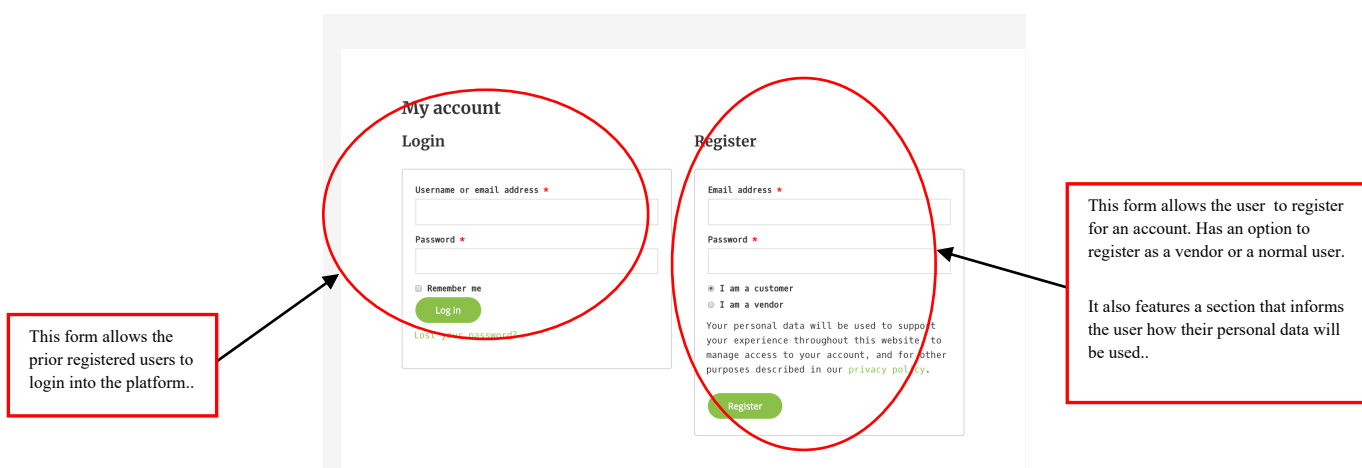


Figure 18 tznini user(my account) account page

4.2.2.7. Admin/vendor dashboard

Figure 4.7 is the admin/vendor dashboard page. After the admin or a user who registered as a vendor logs into the platform, they are led to the dashboard where they can set up their store and view different information about their store.

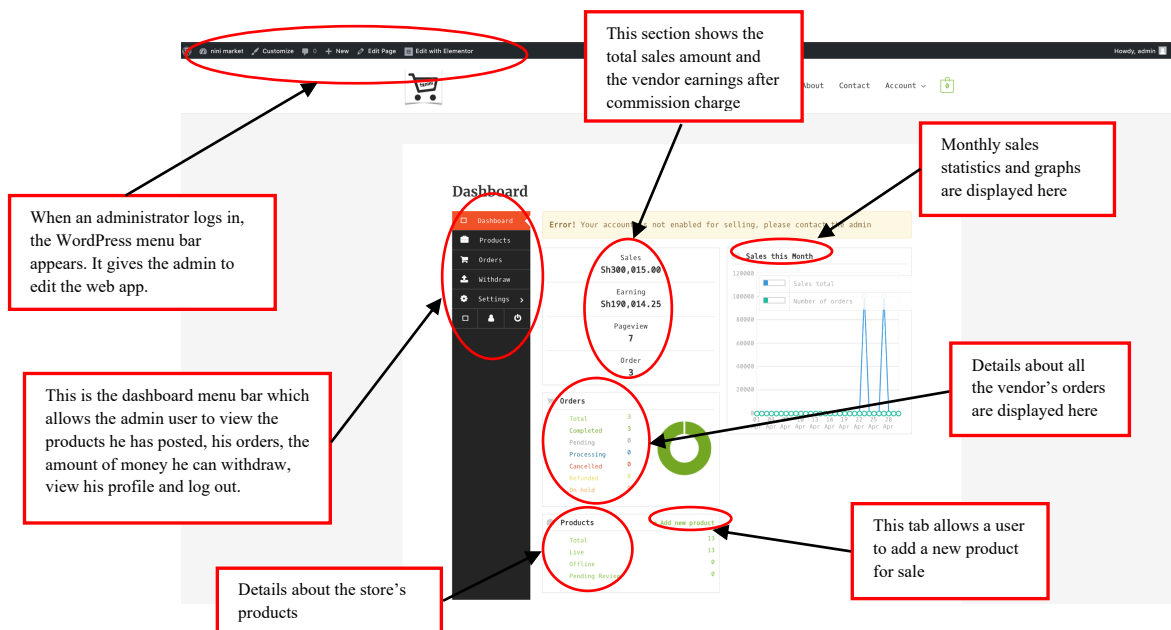


Figure 19 tznini admin/vendor dashboard page

4.2.2.8. Dashboard settings

The dashboard settings page as illustrated on Figure 4.8 allows the vendor to customize their store by adding the store name, banner and other settings.

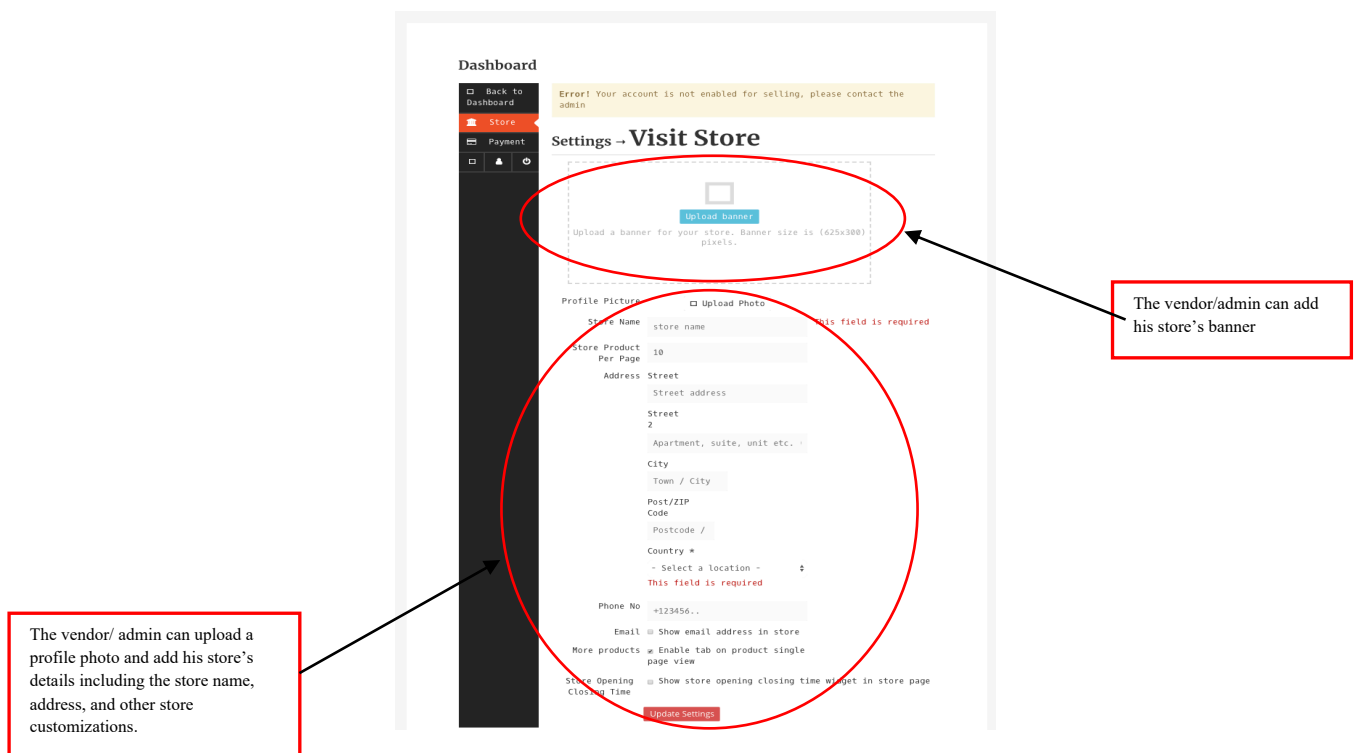


Figure 20 tznini dashboard settings page

4.2.2.9. User account page

The user account page shown on Figure 4.9 is a simpler version of a dashboard page. All users including non-vendors can view this page to manage their accounts as well as log off from the platform.

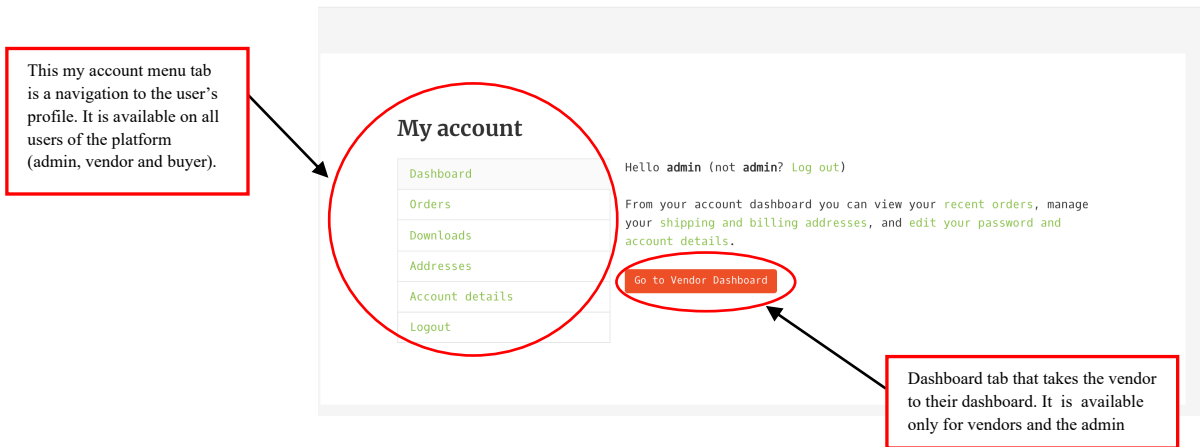


Figure 21 tznini user account page

4.2.2.10. Product view

From the products page if a user clicks on a product, they are led to the page shown on Figure 4.10. It allows the user to view more details about the product such as the reviews, magnify the product, add the product to the cart and more as described below.

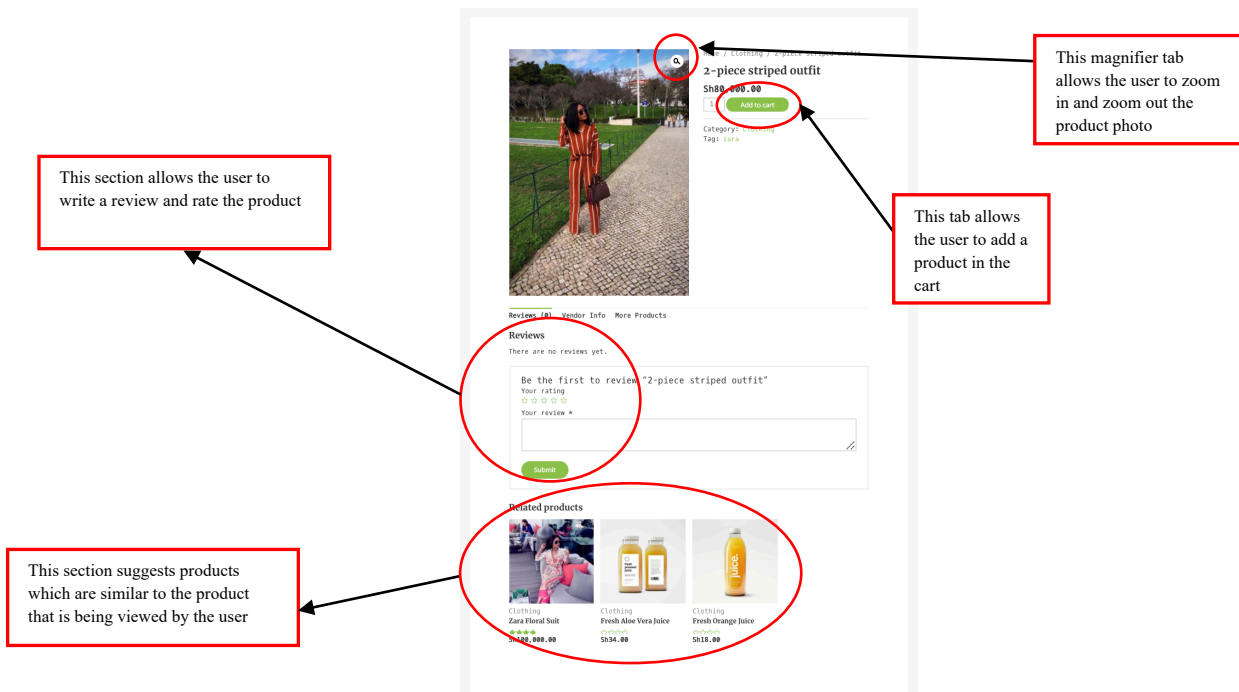


Figure 22 tznini single product view page

4.2.2.11. Cart View

If the user adds a product to the cart then this is how the cart page (Figure 4.11) looks like. The user can add the quantity of the product selected, reduce or even delete the product from the cart completely.

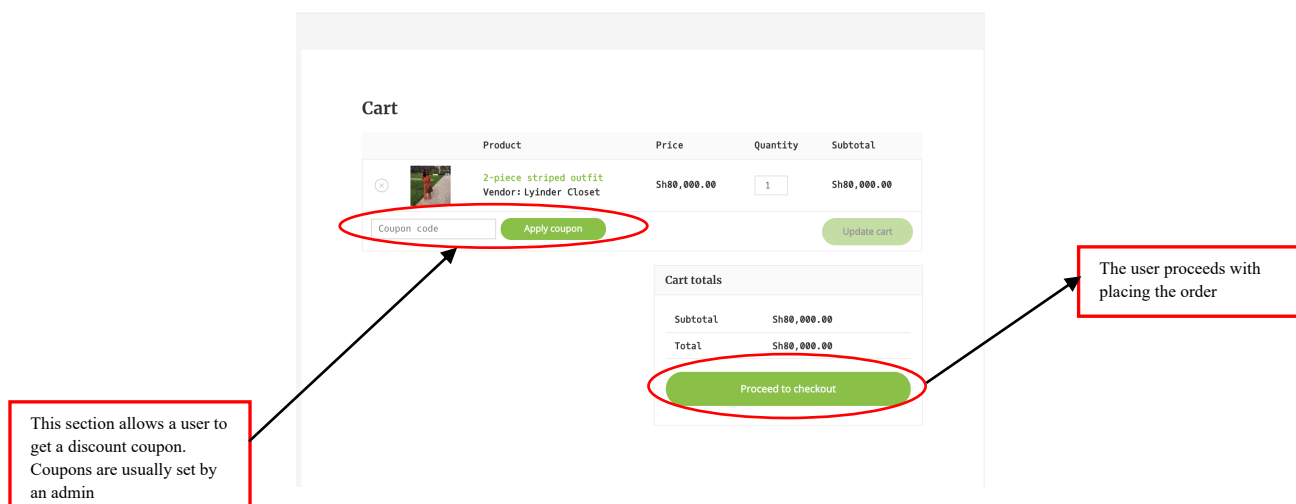


Figure 23 tznini cart view page

4.2.2.12. Checkout

If the user decides to purchase the product from the cart and clicks on the checkout tab is then led to this checkout page as shown on Figure 4.12, where he can add details about how he wants the purchase to be delivered to him, method of payment and confirming the order. The user has the chance to cancel the process before clicking on the place order tab.

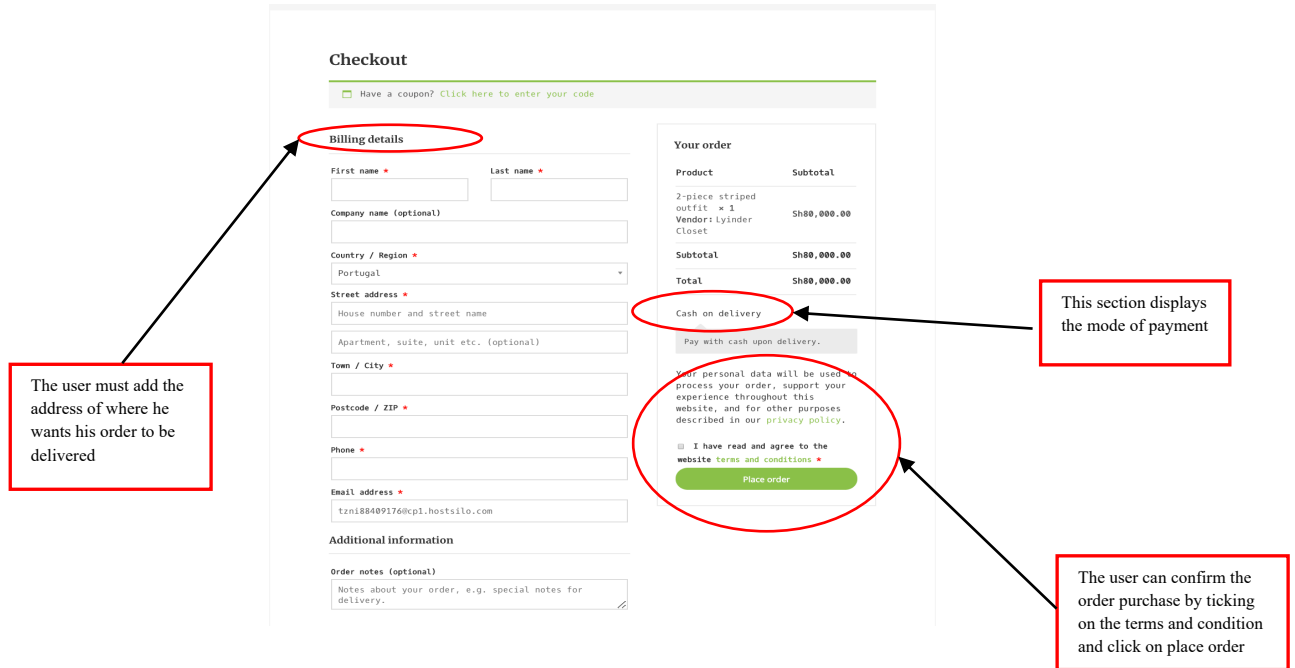


Figure 24 tznini checkout page

4.2.2.13. Order details

After placing the order at the checkout page, the user is then directed to the page illustrated by figure 4.13, where he is informed that the order has been accepted and receives his order details. This serves as a receipt.

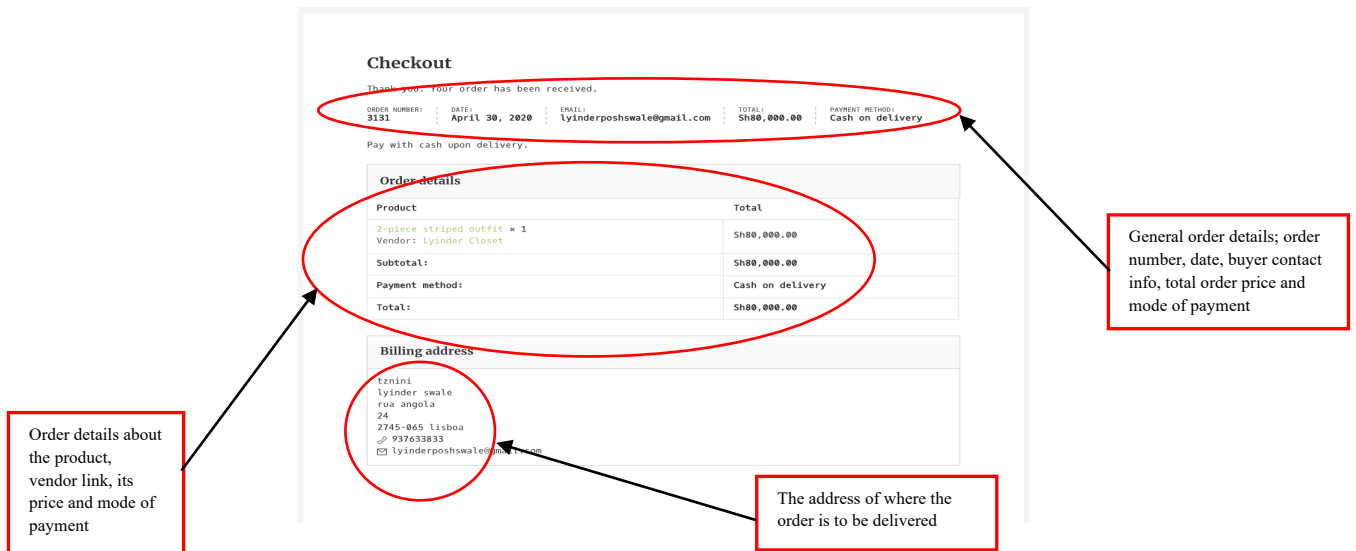


Figure 25 tznini order details view

5. Validation Scenario

5.1.Scenario Purpose, Stakeholders and Equipment

This is the scenario that describes the use of tznini marketplace by the Kariakoo market stakeholders. The stakeholders can be categorised into three groups, namely:

- I. **Wholesalers** who can be further categorised into two groups:
 - a. Importers who bring products in bulk from Asia, Europe or other parts of the world. They normally sell clothing, cosmetics, home and electronic appliances, furniture and more products of the kind, and
 - b. Exporters who mostly are farmers bringing farm produce from other parts of the country to Kariakoo market either for exports or selling it in bulk to other retailers at the market.
- II. **Retailers**, these are vendors who own shop spaces in Kariakoo market. They are also categorized into two:
 - a. Those who import their products either from abroad or upcountry, and
 - b. Those who buy their inventory from the wholesalers.
- III. **Buyers** who are also categorised into two groups;
 - a. Final consumers who buy products for their own personal use, and
 - b. Remote retailers who are either from other parts of the country, from neighbourhood countries like Kenya, Congo or Uganda or own their shops in the suburbs of Dar es salaam who buy products in order to go and sell to their areas.

For the purpose of this paper, the wholesalers and the retailers will be referred to as vendors and all the buyers will be referred to as buyers. However, the retailer's category b will be referred to both as vendors and buyers interchangeably due to the overlapping of their activities and the term user(s) is used as a term that describes both the vendors and buyers collectively.

Equipment that can be used to access tznini marketplace is any mobile phone, computer and tablet with Safari, Google Chrome and Mozilla Firefox browsers.

5.2. Validation Instance

For the platform to be operational, the first users must be the vendors. The following describes the scenarios that take place in tznini marketplace.

User starts browser and types URL of tznini marketplace. The users know the URL through the marketplace brand awareness and marketing which is described on the platform's digital marketing strategy.

The EM application is hosted in the cloud and created and designed using WordPress platform through a web hosting service. It is a 3-tier application consisting of a front-end user interface, business logic layer and the data layer.

User-end layer was designed using an e-commerce WordPress ASTRA template. The template automatically installs the WooCommerce plugin that supports e-commerce activities. The business logic layer was created by installing WordPress plugins which are programmed by PHP language. Dokan which is an e-commerce marketplace plugin was added in order to achieve the EM feature. Other added plugins are Astra widgets for adding widgets, Elementor plugin which is a drag & drop page builder, Elementor – Header, Footer & Blocks for creating custom headers and footers with Elementor and Starter Template Plugin which provides the library of ready-made templates. The data layer that contains the MySQL database and is operated through phpMyAdmin DBMS on the cPanel services form the web hosting platform.

Since TZNINI Marketplace collects users personal data during account registration and purchase processes, it is running through an SSL secure web and has a SSL certificate which was issued through the domain name registrar.

All devices used to access the platform are linked to the database services stored in a cloud server that allows easier access of data from every device that has internet access.. The EM application tries to facilitate the communication between buyers and sellers for making arrangements of orders and the subsequent physical delivery of products.

6. Conclusion

In the first chapter of this document several problems were identified. Namely:

- *The fact that most sellers are not aware of e-commerce benefits;*
- *The inability for sellers to operate an online store since it is it costly and requires constant maintenance;*
- *It is difficult to build and maintain trust between individual online vendors and buyers in terms of the quality and genuineness of the product, price manipulation, the payment security and other fraud related factors; and,*
- *Shipping as a challenge for sellers due to the address system in Tanzania and the cost of shipping.*

In this way, regarding the problem context and following the DSR methodology, several research goals were defined as follows:

RG 1 – Proceed with literature review, in concrete analyzing related topics such as: electronic business, electronic marketplace, approaches to building an online marketplace platform, WordPress as marketplace web application solution;

RG 2 – Define an architecture framework in order to solve the problem. Namely, defining the system domain model, system main use cases (including system users);

RG 3 – Select and analyze a third-party data model to support the system prototype;

RG 4 – Install and test the application which full fits the needed features;

RG 5 – Full describe the prototype user interface; and,

RG 6 – Apply the solution to a concrete validation scenario describing its purpose, stakeholders and equipment.

Thus, several tasks were developed in order to accomplish the referred research goals. Namely:

RG 1 – The literature review, was presented in Chapter 2;

RG 2 – The architecture framework, system domain model, and system main use cases were presented in Chapter 3;

RG 3 – Third-party data model to support the system prototype was selected and analyzed in Chapter 3;

RG 4 – A prototype was installed and tested as described in Chapter 4;

RG 5 – The prototype user interface was presented in Chapter 4; and,

RG 6 – A concrete validation scenario with a purpose, stakeholders and equipment description was provided in Chapter 5.

In this way, as a preliminary conclusion, all the research goals were accomplished as showed. Regarding future work, the aim is to develop new digital products such as a mobile application to support delivering processes regarding the business model.

7. Bibliography

- McKeown, P. (2009). *Information Technology and the Networked Economy*. Global text.
- Kollmann, T. (2013). E-business. *Basics of electronic business processes in the net economy 3rd*.
- Laudon, J. P., & Laudon, K. C. (2012). *Management Information Systems: Managing Digital firms, 13 Edition*. Pearson.
- Coppel, J. (2000). E-COMMERCE . *Impacts and Policy Challenges* .
- Yen, B., & Ng, E. O. (2003). The Impact of Electronic Commerce on Procurement. *Journal of Organizational Computing and Electronic Commerce*, 167-189.
- Croom, S., & Brandon-Jones, A. (2007). Impact of E-procurement: Experiences from implementation in the UK public sector. *Journal of Purchasing and Supply management*, 294-303.
- Presutti, W. D. (2003). Supply Management and E-procurement: Creating value added in the supply chain. *Industrial Marketing Management*, 219-226.
- Knudsen, D. (2003). Aligning Corporate Strategy, Procurement Strategy and E-Procurement Tools. *International Journal of Physical Distribution & Logistics Management Vol. 33*, 720-734.
- Grönlund, Å., & Horan, T. A. (2005). Introducing e-Gov: History, Definitions, and Issues. *Communications of the Association for Information Systems Vol. 15*.
- WorldBank. (2002, 11 1). The E-government Handbook for Developing Countries. *The E-government Handbook for Developing Countries: A Project of InfoDev and the Center for Democracy and Technology*. The WorldBank Group.
- Khan, A. G. (2016). Electronic Commerce: A Study of Benefits and Challenges in Emerging Economy. *Global Journal of Management and Business Research: Economics & Commerce* .
- Flynn , A., & Purchase, S. (2001). *Perception of Barriers to E-commerce*.
- Zaeid, A. N. (2012). Barriers to E-commerce Adoption in Egyptian SMEs. *International Journal of Information Engineering and Electronic Business*.

- MacGregor, R. C., & Vrazalic, L. (2005). A Basic Model of Electronic Commerce Adoption Barriers. *Journal of Small Business and Enterprise Development*, 510-527.
- Heung, V. C. (2003). Barriers to Implementing E-commerce in the Travel Industry: A Practical Perspective. *International Journal of Hospitality Management*, 111-118.
- Darch, H., & Lucas, T. (2002). Training As an E-commerce Enabler. *Journal of Workplace Learning*, 148-155.
- Duan, Y., Mullins, R., Hamblin, D., Stanek, S., Sroka, H., Machado, V., & Araujo, J. (2002). Addressing ICTs Skills Challenges in SMEs: Insights From Three Country Investigations. *Journal of European Industrial Training*, 430-441.
- Bode, S., & Burn, J. (2002). Strategies for Consultancy Engagement for E-business Development - A Case Analysis of Australian SMEs. In S. Burgess, *Managing Information Technology in Small Business Challenges and Solutions* (pp. 227-245). IGI Global.
- Kaynak, E., Tatoglu, E., & Kula, V. (2005). An Analysis of the Factors Affecting the Adoption of Electronic Commerce by SME. *International Marketing Review*, 623-640.
- Efendioglu, A. M., & Yip, V. (2004). Chinese Culture and E-commerce: An Exploratory Study. *Interacting with Computers*, 45-62.
- Love, P. E., Irani, Z., Li, H., Cheng, E. W., & Tse, R. Y. (2001). An Empirical Analysis of the Barriers to Implementing E-commerce in Small-Medium Sized Construction Contractors in the State of Victoria, Australia. *Construction Innovation*, 31-41.
- Bakos, J. Y. (1991). Information Links and Electronic Marketplaces: The Role of Interorganizational Information Systems in Vertical Markets. *Journal of Management Information Systems*, 31-52.
- Segev, A., & Beam, C. (1999). Brokering Strategies in Electronic Commerce Markets. *EC'99: Proceedings of the 1st ACM Conference on Electronic Commerce* (pp. 167-176). ACM.

- Schmid, B., & Stanoevska-Slabeva, K. (1998). Knowledge Media: An Innovative Concept and Technology for Knowledge Management in the Information Age. *Beyond Convergence, 12th Biennial International Telecommunications Society Conference*. Stockholm .
- Kaplan , S., & Sawhney, M. (2000). E-Hubs: The New B2B Marketplaces. *Harvard Business Review*, 78(3), 97.
- Dai, Q., & Kauffman, R. J. (2000). To Be or Not To B2B? An Evaluative Model for E-procurement Channel Adoption. *5th INFORMS Conference on Information Systems and Technology*. Carlspon School of Management.
- Bradley, D. B., & Peters, D. (1997). Electronic Marketplaces: Collaborate if you want to compete. *42nd World Conference International Council for Small Business*. San Francisco.
- Kambil, A., & Van Heck, E. (2002). *Making Markets: How Firms Can Design and Profit from Online Auctions and Exchanges*. Harvard Business Review Press.
- Korper, S., & Ellis, J. (2000). *E-commerce Book: Building the E-Empire, 2nd Edition*. Academic Press.
- Grieger, M. (2003). Electronic Marketplaces: A Literature Review and a Call for Supply Chain Management Research. *European Journal of Operational Research* 144(2), 280-294.
- Kamel, S. (2006). *Electronic Business in Developing Countries: Opportunities and Challenges*. Idea Group Inc.
- Vidyapeeth, T. K. (2010). *Introduction to E-commerce*.
- Messenlehner, B., & Coleman , J. (2019). *Building Web Apps with WordPress: WordPress as an Application Framework, @nd Edition* . O'Reilly.
- Bodo, S. (2003). *Website Development & Hosting*. Harare, Zimbabwe: Fontline Electronic Publishing.
- Zhang, H. (2013). Architecture of Network and Client-Server Model . *Journal of Advances in Electrical Engineering Systems*.

- Kambalyal, C. (2010). *3-Tier Architecture*. Retrieved from channukambalyal.tripod.com:
channukambalyal.tripod.com/NTierArchitecture.pdf
- WordPress Org. (2020). *Theme Handbook*. Retrieved from WordPress.Org:
wordpress.org/theme_handbook
- Choudhury, N. (2014). World Wide Web and Its Journey from Web 1.0 to 4.0 .
International Journal of Computer Science and Information Technologies 5(6),
8096-8100.
- Cazañas, A., & Parra, E. (2017). Strategies for Mobile Web Design . *Enfoque UTE*,
7(1), 344-357.
- Lestari, D., Hardianto, D., & Hidayantp, A. (2014). Analysis of User Experience
Quality on Responsive Web Design from its Informative Perspective .
International Journal of Software Engineering an its Applications 8(5), 53-62.
- Majid, E., Kamaruddin, N., & Mansor, Z. (2015). Adaptation of Usability Principles in
Responsive Web design Technique for E-commerce Development. *International
Conference on Electrical Engineering and Informatics (ICEEI)*, (pp. 726-729).
Denpasar, Indonesia.
- Rashid, B. (2017, October 27). *5 Essential Reasons You Should be Using Responsive
Web Design Now*. Retrieved from forbes.com:
<https://www.forbes.com/sites/brainrashid/2017/06/13/5-essential-reasons-and-benefits-why-you-should-be-using-a-responsive-web-design-now/#1a1d1f0517c9>
- Winkless, C. (2015). *Using Responsive Web design to Implement a Mobile-First Digital
Strategy* . Retrieved from madebykind.com:
<https://www.madebykind.com/thinking/using-responsive-web-design-to-implement-a-mobile-first-digital-strategy>
- Baker, S. (2014). Making it Work for Everyone: HTML5 and CSS Level 3 for
Responsive, Accesible Design on your Library's Web Site . *Journal of Library
& Information Services in Distance Learning* 8(3-4), 118-136.
- Ogala , E., & Sylvester , A. O. (2019). Concept of Web Programming: An Overview of
PHP. *International Journal of Scientific and Engineering Research* 10(12).

- WordPress Org. (2020). *Plugins*. Retrieved from WordPress Org:
<https://www.wordpress.org/codex/what-is-a-plugin>
- Stroe, I. (2011). MySQL Databases as Part of the Online Business, Using a Platform Based on Linux, 2(3). *Database Systems Journal* .
- Schäferhoff, N. (2018). The Decision Maker's Guide to E-commerce on WordPress . In S. N, *The Ultimate Guide to E-commerce on WordPress* (p. Chapter 1). Torque Magazine.
- Puetz, C. (2005). *The Web Hosting Manager*. Lulu.com.
- Stonewell, A. (2015). *Everything You Need To Know About Web Hosting*. Apple Books.
- Goyal, S. (2013). Software as a Service, Platform as a Service, Infrastructure as a Service - A Review. *International Journal of Computer Science & Network Solutions* 1(3), 53-67.
- Srivastav , M. K., & Nath, A. (2016). Web Content Management System. *International Journal of Innovative Research in Advanced Engineering (IJIRAE)* 3(3), 51-56.
- Hevner, A. R., March , S. T., Park, J., & Ram , S. (2004). Design Science in Information System Research. *MIS Quarterly* 28(1), 75-105.

Annex

A. Back-Office UI Screenshots

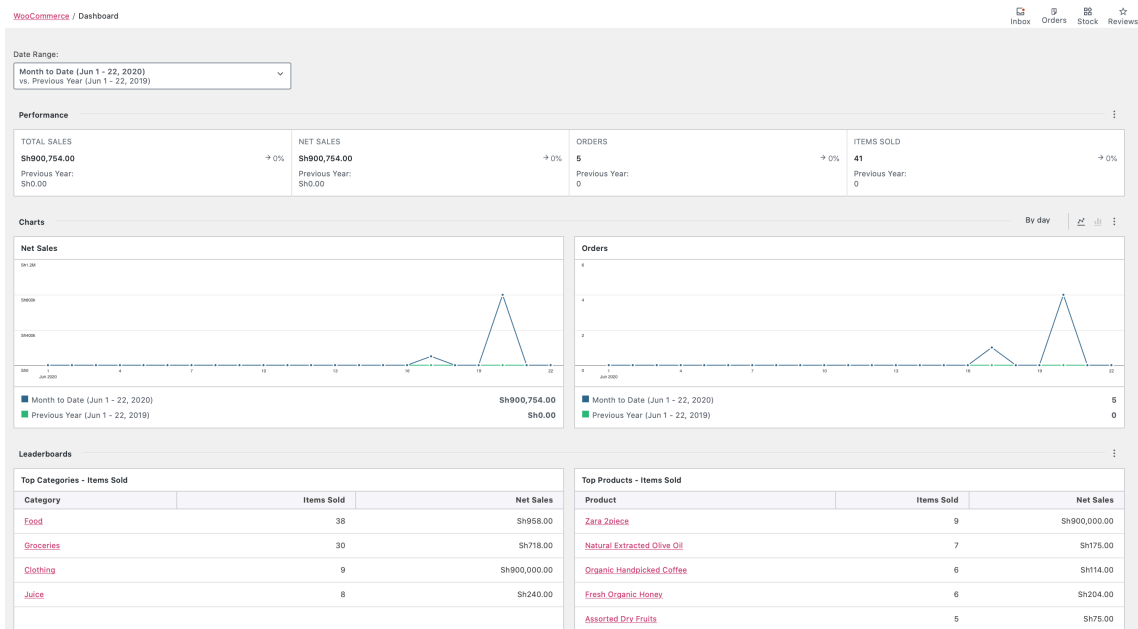


Figure 26 WooCommerce Dashboard

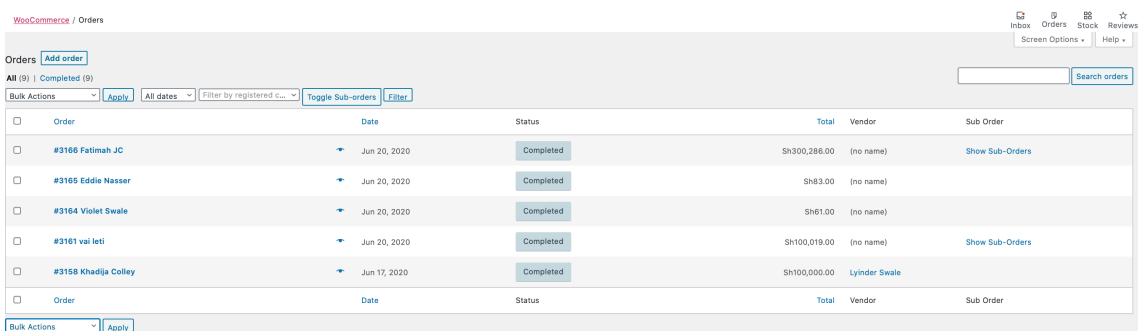


Figure 27 WooCommerce Orders screen

KARIAKOO GOES ONLINE: Placing Traditional Tanzanian Market on an Online Marketplace Platform – Sistemas e Tecnologias da Informação

WooCommerce / Products

Inbox Orders Stock Reviews
Screen Options Help

Products [Add New](#) [Import](#) [Export](#)

All (15) | Published (15) | Sorting

Bulk Actions [Apply](#) | Select a category | Filter by product type | Filter by stock status [Filter](#)

Search products

<input type="checkbox"/>	Name	SKU	Stock	Price	Categories	Tags	Date	Author
<input type="checkbox"/>	Cantu	-	In stock	Sh38,000.00 Sh32,000.00	Hair	cantu, fatima, hair, natural hair	Published 2020/06/20	fatimahisa23
<input type="checkbox"/>	Gucci Mules	-	In stock	Sh800,000.00 Sh720,000.00	Shoes	fatima, Gucci, Mules, Shoes, Zara	Published 2020/06/20	fatimahisa23
<input type="checkbox"/>	Zara 2piece	-	In stock	Sh100,000.00	Clothing	Zara	Published 2020/06/19	Lyinder Swale
<input type="checkbox"/>	Organic Handpicked Coffee	-	In stock	Sh19.00	Groceries	-	Published 2018/06/20	
<input type="checkbox"/>	Assorted Dry Fruits	-	In stock	Sh15.00	Groceries	-	Published 2018/06/20	
<input type="checkbox"/>	Fresh Orange Juice	-	In stock	Sh18.00	Juice	-	Published 2018/06/20	
<input type="checkbox"/>	Organic Lentils	-	In stock	Sh15.00	Groceries	-	Published 2018/06/20	
<input type="checkbox"/>	Fresh Aloe Vera Juice	-	In stock	Sh34.00	Juice	-	Published 2018/06/19	
<input type="checkbox"/>	Fresh Organic Honey	-	In stock	Sh34.00	Groceries, Juice	-	Published 2018/06/19	
<input type="checkbox"/>	Natural Extracted Olive Oil	-	In stock	Sh34.00 Sh23.00	Groceries	-	Published 2018/06/19	
<input type="checkbox"/>	Assorted Coffee	-	In stock	Sh35.00	Groceries	-	Published 2018/06/19	
<input type="checkbox"/>	Black Lentils	-	In stock	Sh34.00	Groceries	-	Published 2018/06/19	
<input type="checkbox"/>	Kidney Beans Organic	-	In stock	Sh38.00 Sh25.00	Groceries	-	Published 2018/06/19	
<input type="checkbox"/>	Organic Pigeon Peas	-	In stock	Sh35.00	Groceries	-	Published 2018/06/19	
<input type="checkbox"/>	Rich Organic Tea	-	In stock	Sh38.00 Sh25.00	Groceries	-	Published 2018/06/19	

Bulk Actions [Apply](#)

15 Items

Figure 28 WooCommerce Products

WooCommerce / Customers

Inbox Orders Stock Reviews

Show: All Customers

Customers Search by customer name [Download](#)

Name	Username	Last Active	Sign Up	Email	Orders	Total Spend	AOV	Country / Region	City	Region	Postal Code
posh_swale	poshylicious	June 22, 2020	June 15, 2020	info@tznzi.com	1	Sh100,000.00	Sh100,000.00	TZ	lisbea	TZ02	1234
Violet Swale		June 20, 2020	—	vixarex339@gmail.com	1	Sh61.00	Sh61.00	TZ	Dar es salaam	TZ02	37128
val leti		June 20, 2020	—	poshylicious@yahoo.co.uk	1	Sh200,038.00	Sh200,038.00	TZ	dar es salaam	TZ02	2745065
Eddie Nasser	ranetehtid	June 20, 2020	June 20, 2020	ranetehtid@gmail.com	1	Sh83.00	Sh83.00	TZ	Knightdale	TZ02	27713
Fatimah JC	ftimahjc	June 20, 2020	June 20, 2020	ftimahjc@gmail.com	1	Sh600,572.00	Sh600,572.00	TZ	Dares Salam	TZ02	POBOX285
	fatimahisa23	June 19, 2020	June 19, 2020	fatimahisa23@gmail.com	0	Sh0.00	Sh0.00				

6 customers 1 average order Sh180,150.80 average lifetime spend Sh180,150.80 average order value

Figure 29 WooCommerce Customers Screen

KARIAKOO GOES ONLINE: Placing Traditional Tanzanian Market on an Online Marketplace Platform
 – Sistemas e Tecnologias da Informação

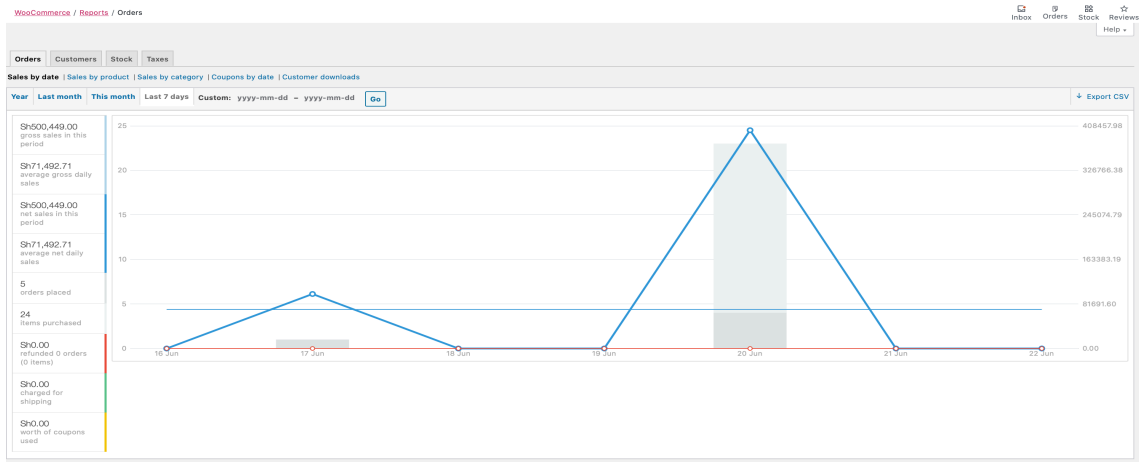


Figure 30 WooCommerce Reports Screen

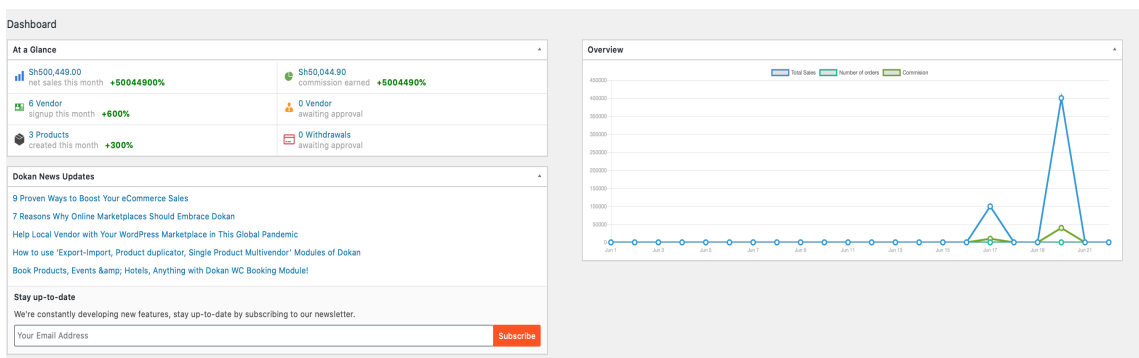


Figure 31 Dukan Dashboard

Store	E-mail	Phone	Registered *	Status
(no name)	fatimahisa23@gmail.com		Jun 19, 2020	ON
Samica Collection	winsarp@yahoo.com	+48708555890	Jun 17, 2020	ON
Sinza	ellenodunga@gmail.com	071501147	Jun 17, 2020	ON
Ellen	lyindercoltd@gmail.com	+25575501147	Jun 15, 2020	ON
Eve	info@tznini.com	+2550000999	Jun 15, 2020	ON
Lyinder	info@lyinder.com	937633833	Jun 15, 2020	ON

Figure 32 Dokan Vendors

B. Database Tables

Table	Action	Rows	Type	Collation	Size	Overhead
admin_notes	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	16.0 KiB	--
admin_note_action	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
announcement	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	48.0 KiB	--
api_keys	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
category	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	16.0 KiB	--
categorymeta	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
category_relationships	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	48.0 KiB	--
category_taxonomy	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
commentmeta	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
comments	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
downloadable_product_permission	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	64.0 KiB	--
download_log	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
links	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	16.0 KiB	--
log	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	16.0 KiB	--
option	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	16.0 KiB	--
order	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
order_item	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
order_itemmeta	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
order_product_lookup	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	48.0 KiB	--
payment_token	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
payment_tokenmeta	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
product	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
productmeta	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
refund	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	48.0 KiB	--
session	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	16.0 KiB	--
shipping_zone	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	16.0 KiB	--
shipping_zone_location	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
shipping_zone_methods	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
tax_rate	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
tax_rate_class	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	16.0 KiB	--
tax_rate_location	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
user	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	16.0 KiB	--
usermeta	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
user_lookup	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
vendor_balance	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--
webhooks	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	16.0 KiB	--
withdraw	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_general_ci	32.0 KiB	--

Figure 33 Tznini Database Tables on phpMyAdmin

KARIAKOO GOES ONLINE: Placing Traditional Tanzanian Market on an Online Marketplace Platform
 – Sistemas e Tecnologias da Informação

Show all | Number of rows: 25 | Filter rows: Sort by key: None

ID	user_login	user_pass	user_nicename	user_email	user_url	user_registered	user_activation_key
4	lyinderposhwale	\$PSBZYF.pfECTHOnUaihydqCyAmAbP/z/	lyinderposhwale	lyinderposhwale@gmail.com		2020-06-15 23:11:07	
2	lyinder.com	\$PSBoGKteJsE.aSFnxTbTLLeEmdHdP24UM/	lyinder	info@lyinder.com		2020-06-15 22:12:32	1592260515:\$PSBxvNn2Sh\$SCq
3	poshylicious	\$PSu2viqdLL15RoKbTtVUV4HJ32hSve.	poshylicious	info@tznini.com	http://tznini.com	2020-06-15 22:40:50	
5	lyinderooldt	\$PSBbhDiXejL6VIMKfPmWzLSEFSv7tkcr/	ellen	lyinderooldt@gmail.com		2020-06-15 23:23:04	
6	ellenodunga	\$PSB9pbMwo/kOipxTAIEAwjyspgQpfc2.	sinzamadukani	ellenodunga@gmail.com		2020-06-17 19:19:51	
7	winsarp2	\$PSBDjsV.JKbTr5dnxJ58eMkklur7nFa1	samica-collection	winsarp2@yahoo.com		2020-06-17 19:35:14	
8	fatimahisa23	\$PSBpccuG1J2GK5KjLJE0qp5uunNSEig.	fatimahisa23	fatimahisa23@gmail.com		2020-06-19 14:31:54	
9	ranetechltd	\$PSB.pDdNCyF.T1aNly7O/z7bzT5V2QPv.	ranetechltd	ranetechltd@gmail.com		2020-06-20 18:18:27	
10	ftimahjc	\$PSB/pu5BM8twD2qiRdafigHIBIK7UvMm0	ftimahjc	ftimahjc@gmail.com		2020-06-20 19:23:39	

Check all | With selected:

Figure 34 Users Table

Show all | Number of rows: 25 | Filter rows: Sort by key: None

id	order_id	seller_id	order_total	net_amount	order_status
1	Browse	2	100000.0000	90000.0000	wc-completed
2	3162	2	100000.0000	90000.0000	wc-completed
3	3163	1	19.0000	17.1000	wc-completed
4	3164	1	61.0000	54.9000	wc-completed
5	3165	1	83.0000	74.7000	wc-completed
6	3167	1	286.0000	257.4000	wc-completed
7	3168	2	300000.0000	270000.0000	wc-completed

Check all | With selected:

Show all | Number of rows: 25 | Filter rows: Sort by key: None

Figure 35 Orders Table

1 > >> | Show all | Number of rows: 25 | Filter rows: Search this table

+ Options				
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
term_id	name	slug	term_group	
1	Uncategorized	uncategorized		0
2	Primary	primary		0
3	Primary (2)	primary-2		0
4	Social Links Menu	social-links-menu		0
5	simple	simple		0
6	grouped	grouped		0
7	variable	variable		0
8	external	external		0
9	exclude-from-search	exclude-from-search		0
10	exclude-from-catalog	exclude-from-catalog		0
11	featured	featured		0
12	outofstock	outofstock		0
13	rated-1	rated-1		0
14	rated-2	rated-2		0
15	rated-3	rated-3		0
16	rated-4	rated-4		0
17	rated-5	rated-5		0
18	Clothing	clothing		0
19	Main Menu	main-menu		0
20	Groceries	groceries		0
21	Juice	juice		0
22	page	page		0
23	section	section		0
24	Food	food		0
25	Beauty	beauty		0

Figure 36 Terms (Categories) table

C. Responsive Design Screenshots

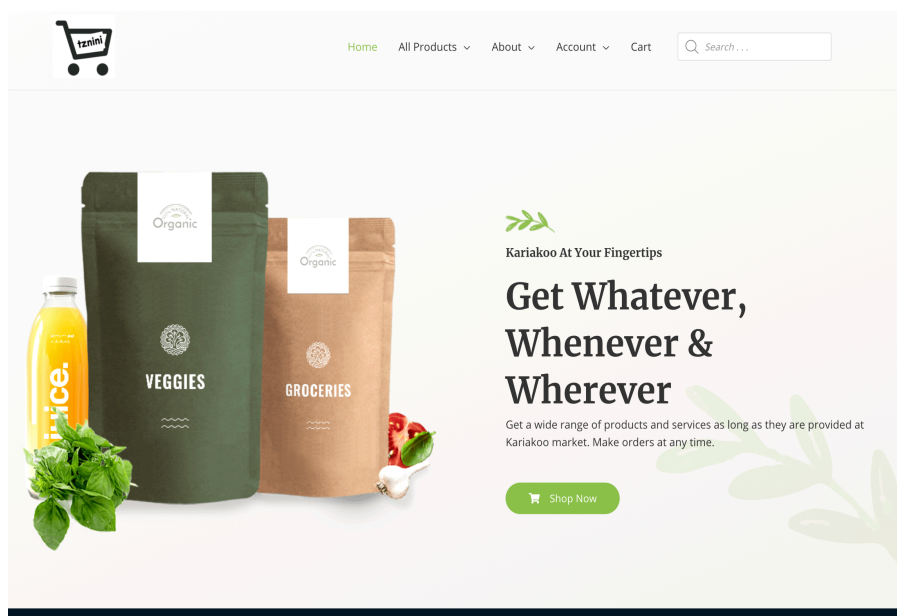
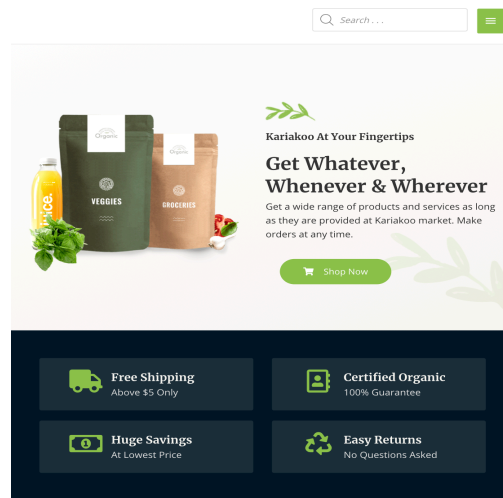


Figure 37 tznini Desktop View



Best Selling Products

Figure 38 tznini Tablet View

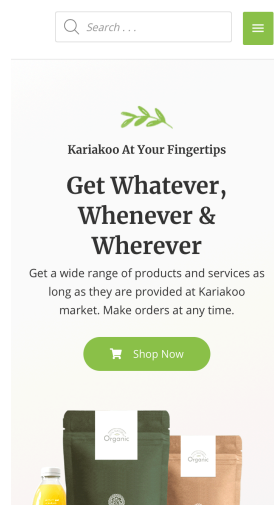


Figure 39 tznini Mobile Phone View

Appendix

