ASHESI UNIVERSITY

A WEB BASED APPLICATION WITH NUMEROUS

FUNCTIONALITIES TO SUPPORT ACTIVITIES OF AFRICAN

POLISTAS IN GHANA

APPLIED PROJECT

B.Sc. Management Information Systems

Sadat Issah

2020

ASHESI UNIVERSITY

A WEB BASED APPLICATION WITH NUMEROUS

FUNCTIONALITIES TO SUPPORT ACTIVITIES OF AFRICAN

POLISTAS IN GHANA

APPLIED PROJECT

Applied Project submitted to the Department of Computer
Science/Information Systems, Ashesi University in partial fulfilment of the
requirements for the award of Bachelor of Science degree Management
Information Systems.

Sadat Issah

2020

DECLARATION

I hereby declare that this applied project is the result of my own original work and that no part of it has been presented for another degree in this university or elsewhere.

Candidate's Signature:

……………………………………………………………………………………………

Candidate's Name:

……………………………………………………………………………………………

Date:

……………………………………………………………………………………………

I hereby declare that preparation and presentation of this capstone type were supervised in accordance with the guidelines on supervision of Dr. Stephan Nwolley Jnr. laid down by Ashesi University.

Supervisor's Signature:

……………………………………………………………………………………………

Supervisor's Name:

……………………………………………………………………………………………

Date:

……………………………………………………………………………………………

i

Acknowledgement

Abstract

Technology has become a core part of most businesses both in the formal and informal sectors. Businesses are leveraging on the potency of technology to improve processes and increase revenue. African Polistas is a polo club here in Ghana having a branch in South Africa, USA and other parts of the world. Currently the club does not have an online presence making it difficult to connect with its members and reach out to prospective members as well. Due to its absence online, the club is unable to sell its merchandise to a wider market. This project focuses on building a web-based application for African Polistas to give it an online presence. Among the features of the application are; e-commerce platform, a chatbot, community chat platform, service subscription and membership registration services. The e-commerce platform is to enable the club make sales of its merchandise online and reach a wider market. The community chat forum is to connect all members of the club so they can network and know each other. The club offers horse riding school services, tourism services, takes part in polo tournaments; registered members of the club should be able to sign up for these services.

Table of Content

Chapter 1: Introduction

1.1   Introduction

In today's more complex and diverse market and economy, businesses are always on the lookout for alternative markets to showcase and sell their products. The emergence of the internet was one of the revolutionary inventions of all time and now undeniably, every aspect of our life depends on the use of the internet. The idea of the internet can be credited to the ARPANET (Advanced Research Project Agency Network) team which was funded by the US Department of Defence in the 1960s. ARPANET originally deployed packet switching which enables computers on a network to communicate with each other, most especially, research institutions funded by Pentagon. Research continued in the field and due to the inefficiencies of packet switching, the Transmission Control Protocol (TCP) was introduced in the 1970s by Robert Khan and Vinton Cerf [1]. From there, there have being series of research and inventions which has led to what we now have as the internet. TCP was then merged with the Internet Address protocol to have TCP/IP. The emergence of e-commerce saw some firms shift completely from operating brick and mortar to operating on the internet whilst others partly trading on the internet. Companies such as Amazon, eBay, Alibaba, and recently Jumia which is mainly operating in the African market. It is prudent noting that, an e-commerce business can take the form of Business to Business; where businesses transact with each other over the internet, Business to Consumer; where businesses trade with customers on the internet and also Government to Business; where the transaction is solely between governments and the business. One interesting thing about the internet market is that, almost all forms of products and services can be traded there at minimal cost. Operating a business online has numerous benefits to the firm; overcoming geographical limitations to reach new customers, gaining new customers through search engine visibility, the minimal cost of operation, customers can

compare various products available online, makes it easier for firms to advertise and give out promotions and many other benefits for both the firms and their clients [2]. Firms engaging in various forms of sporting businesses have also not being left out in taking advantage of e-commerce to market what they do. Sports ranging from Soccer, Cricket, Basketball, Hockey, Polo and many others. With just a simple google search, one is likely to find numerous platforms trying to reach out to a wider market to sell various merchandise. Examples are spikeball.com, condorcycles.com, purecycles.com, and even DStv Africa. Shifting focus to Polo sports which most people refer to as "The Gentleman's Sports" or "The Kings Sports". The game is played using horses with teams of two and each team having four players each. The players sit on the horseback using a long and flexible mallet handle to push a ball on the grass trying to score [3]. With this sport, members interested usually affiliate themselves with clubs, from such clubs they can get horses for training and other services since horses are expensive to buy and keep. Some of the polo clubs around the world have found their way to pin their internet presence whilst some are still not taking advantage of the internet to expand their network. An example is the Africa Polistas here in Accra with a branch in South Africa which is not having an online presence. The clubs already having an online presence are also not taking full advantage of what an online presence has to offer. Most of the sites I visited only have information about the clubs, some images and videos of recent tournaments held or played and some few announcements if there is any. Examples are ladolfina.com, ellerstina.com among others. Any polo club having an online presence has a wide range of opportunities. The club can have an interactive portal that engages members of the club or visitors to share thoughts and ideas regarding the sport. They can also advertise and sell Polo merchandise on the platform and even list some of the horses they have which are for sale. Also, if they provide horse riding training services, they can advertise that as well for members of the public to subscribe. In a nutshell, this project is going to focus on how to digitize the

operations of a polo club here in Ghana with a branch in South Africa called Africa Polistas and give it an online presence.

1.2   Problem Statement

Africa Polistas is a registered polo club here in Accra and has a branch in South Africa with members spanning across the globe. The club has members in Argentina, England and the USA. Currently, the club does not have a reliable database of its members making it difficult to keep in touch with members across the world. The club does not also have an online presence limiting its market reach considering the numerous services they render. Their services include horse riding school, tour services, taking part in global polo tournaments among others. The club also engages in selling horses and other polo merchandise. Currently, their reach is limited only to friends and close associates. This is due to their inability to reach out to broader market space.

1.3   Proposed Solution

What Africa Polistas needs is an integrated web application which will allow them to sell what they do to diverse group of people whilst keeping and maintaining a good relationship with their already existing client base. On the application, they will have various sections purposely discussing the activities they engage in and the products and services they offer. There will be a portal that allows members to subscribe to the various horse-riding training packages the club has. Persons who are interested in becoming members of the club should be able to make a request on the platform. To connect all members of the club, the club will want to have a secured chat forum for its registered members. This they believe will help their members to keep in touch and network. The chat forum will in part help members to share reviews of their activities with each other.

1.4 **Motivation**

My motivation for undertaking this project is in two folds. One is due to my interest in Polo games and horse riding. My second motivation lies in the fact that, I am working on this for a client and solving an issue facing the club. I am rest assured that, when well executed, will be implemented and used by the organization. This project will allow me to bring together the bits and pieces of skills I have acquired over the years into one unique final product for the client.

1.5 **Technical Background Information**

Packet Switching: It is a technique whereby before a piece of data is transferred to a destination, the data is broken into suitable sizes and sent across various channels. This is expected to reduce latency and increase delivery time. The data sent is then brought together by networking devices and delivered to the destination address.

TCP/IP: it is a collection of protocols that specify how devices should communicate and interchange data with each other on networks. It determines how data should be divided, the routing channels to use and how data should be delivered to the destination.

1.6   RELATED WORK

1.6.1   Trustworthiness in E-commerce

There is a lot of contention and several types of research made regarding what contributes to the success of e-commerce platforms and what constitutes failure as such. Many authors and industry players have attributed it to the usability of the system. In this article, the authors indicated that, yes, the usability of the site is key for the success of an e-commerce site but that alone is not enough since there are numerous sites with great usability features which due to the lack of trust, the sites fail. They compared sites with great usability features and sites with trustworthy features and came up with the following success factors; minimizing consumer risk, building long term relationships and strong reputation, and utilizing third parties [4]. They realized that, sites with good usability features but lack the above mentioned tend to fail. Hence sites without trustworthiness will fail. The author does not completely refute the idea of usability but believes that the two are needed to achieve a successful e-commerce platform [4]. This piece is relevant to my project because it gives me an overview of what I should be considering when building my application. From my side as a developer, I believe that, there is less I can do in building the trustworthiness of the system. The issue of trust mainly comes from the firm in its dealings with its clients. Usability will be my focus.

1.6.2   Chatbots and The World of HCI

Research for most parts has focused on building interfaces to enhance human-computer interactions. A lot has changed lately with the advent of natural language processing where interfaces must be designed to support digital interactions [7]. This is evident in recent developments of bots, both text messaging format and voice messaging format. Technology giants like Google, Facebook, Microsoft have all taken to bots to enhance their services. Machine agents in the form of bots can improve user experience and help in user data

generation. The article identifies some of the benefits of a chatbot as helping to improve customer services, improve engagement with customers, data collection and analytics [7]. This article is relevant to my project because as part of the project, a chatbot will be implemented to aid user interaction on the platform.

Chapter 2: Requirements

2.1   Overview

My applied project is a direct request from African Polistas for me to build a web application with numerous functionalities for the club. Following the initial request which contained some key functionalities of the system, I made a field trip to the club where I extensively discussed the project with the managers and some of the employees of the club. We discussed their expectations, what could constitute the success of the project and the major functionalities of the web application. I took the chance to further understand the operations of the club through observation and what their challenges are. Aside from the site visit, I have had several engagements with Harold who is the man in charge of the activities of the club. Through these engagements, I developed a much better understanding of the project to craft the functional and non-functional requirements, use case tables and diagrams. I as well checked on the websites of a few polo clubs across the globe to have a fairer understanding of what they have in common and what they are lacking.

The sections that follow will elaborate on the functional requirements, non-functional requirements, use case diagrams and the various users of the application and how they will interact with it.

The club wants to maintain separate portals for the South African club and a separate application for the Ghana club. From the requirements gathering, it was realized that, the functionalities of the clubs seem different. Below I will categorize the functionalities based on the club.

2.2 Key Features of the Project

South African Club:

- Riding school portal

- polo merchandise selling

- Horse selling

- Registration of members to the portal

- Chat forum for all registered members of the club

Ghana Club:

- Riding School

- Weather forecast and prediction

- Selling merchandise

- Selling of horses

- Registration of members to the portal

- Chat forum for all registered members of the club

2.3 Users/Actors

The platform will be used by three key people. The Administrator who will oversee updating the system and reviewing of the registration request by members. Another actor is a registered member of the club who will be able to, aside all other functionalities chat with other members in the chat platform. Another is a passer-by who is new to the system and just want to see what is available on the site.

2.3.1 System Administrator

The system administrator will be responsible for controlling a couple of things on the platform. The administrator will be responsible for controlling the e-commerce site from the dashboard; these include adding new products, confirming purchase and shipping of products etc. The system will accept user requests from members who want to be part of the club. Upon satisfactory review of the request, the administrator will have to approve and to allow the user to be part of the main platform of members. The administrator will be responsible for maintaining the chat platform.
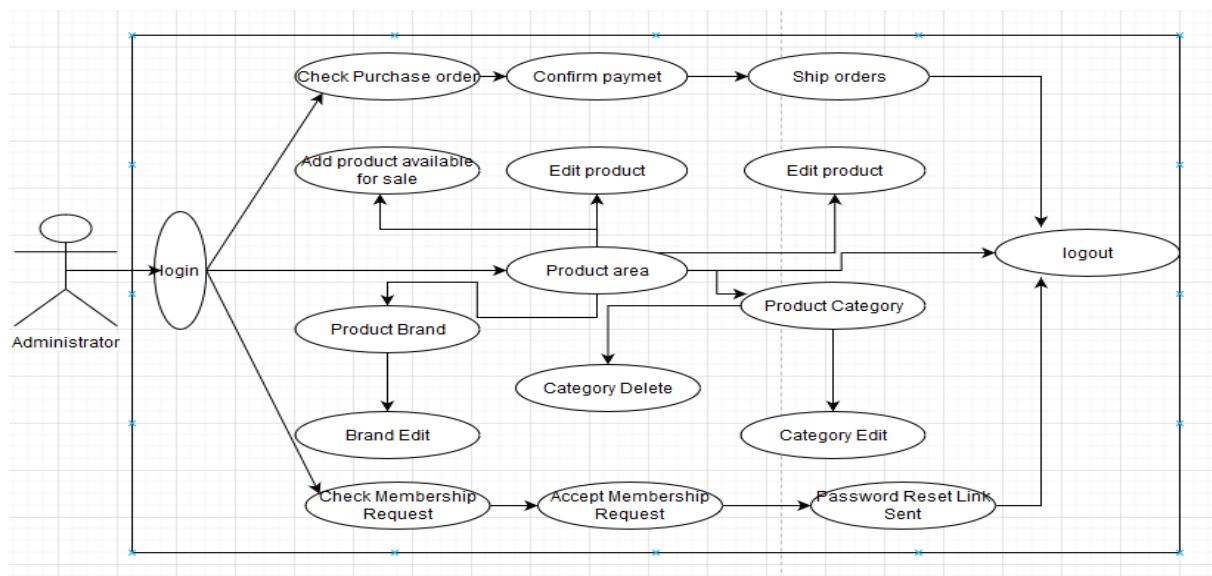

Figure 7.0 showing use case diagram of system administrator.

Use case table for Administrator

Table 2.1 showing use case table for system administrator.

Use Case: Login	Actor: System Administrator. Requirement ID: SA001
Stakeholder	System Administrators
Description	This is to demonstrate administrator login.
Goal	Validate, authenticate and authorize the administrator when logged in.
Success Measurement	Correct credentials are accepted by the system and access given whilst improper credentials are rejected by the system.
precondition	Login credentials have been populated in the database already.
Trigger	The application can only be accessed via the web through a browser.
Relationship	When the administrator is successful with login, an admin interface shows to allow the administrator to do what he/she wants.
Event Flow	The administrator clicks on the login button to log into the system. 1. Login page displays. 2. administrator enters log in details using email and a password. 3. Dashboard is displayed.
Use Case: Accept user registration	Actor: System Administrator Requirement ID: SA002
Stakeholder	The system administrator
Description	This use case is to detail how the system administrator will accept membership request.
Goal	Verify details of the applicant before accepting registration.
Success measurement	Administrator approves user registration depending on the credentials given. A URL link sent to help the user set login credentials.
Precondition	User should have filled an online form requesting to be a member of the club. Also, the administrator should be logged in.
Trigger	The administrator clicks on the approve button.
Relationship	When the member is approved, an email is sent to the user to change password. The

	user status also changes to approved on the admin dashboard
Event flow	1. Administrator successfully logs into the system. 2. Administrator clicks on users. 3. Administrator checks details and background information. 4. User is accepted or rejected; an email is sent when the user is accepted.
Use case: Revoke membership	Actor: System administrator Requirement ID: SA003
Interested stakeholder	System administrator
Description	System administrator can remove members who cease to be part of the club.
Goal:	Successful removal of members who cease to be part of the club.
Success measurement	When the removed member is unable to login into the system.
Precondition:	Administrator logs into the dashboard.
Trigger:	Administrator clicks on the remove button.
Event flow:	1. Administrator logs into the system. 2. Searches through users list to identify the member 3. Clicks on the remove button to successfully remove member. 4. Notification shows remove successfully.
Use case: Chat platform	Actor: System Administrator Requirement ID: SA004
Interested stakeholders	System administrator
Description:	System administrator should be able to join the chat platform to interact with members of the club.
Goal:	Administrator should be able to have a fluid chat with members of the club on the platform.
Success measurement:	Administrator should be able to successfully chat with members on the platform
Precondition:	Administrator should have logged into the system and opened the chat portal.
Trigger:	The administrator starts reading and typing of message.

Event flow:	1. Administrator logs in. 2. Begins to read messages and typing messages.
Use case: add products and services	Actor: system administrator Requirement ID: SA005
Interested stakeholders:	System administrator
Description:	System administrator should be able to update the products and services in the products and services section of the platform since there will be buying and selling done on the platform. The administrator will specify product brands and categories. Administrator should be able to check orders, confirm if payment is done and make shipment of product.
Goal:	Able to successfully update the products and services page with product and service details. Check orders and ship products purchased.
Success measurement:	Administrator can add products which successfully displays on the product page. Administrator can check product orders and ship a product
Precondition:	Administrator logs into the system.
Trigger:	Add product button is clicked.
Event flow:	1. Product details form is filled. 2. Image of the product is uploaded 3. Add button is clicked and product added successfully Orders: 1. Check orders table 2. Confirm payment is done 3. Approve and ship product. Create Brands and Categories: 1. Click on either brands and categories. 2. Create or edit or delete a brand or category.
Use Case: Log out	Actor: System Administrator. Requirement ID: SA006
Stakeholder	System Administrators
Description	This is to demonstrate administrator log out.

Goal	Able to successfully end the session and log administrator out.
Success Measurement	Administrator is successfully logout and page redirects to login page.
precondition	Administrator already logged in.
Trigger	The application can only be accessed via the web through a browser.
Relationship	When the administrator is successful with the logout, he or she is redirected to the log in page.
Event Flow	The administrator clicks on the log out button to log out.

2.3.2   User/Members of the Club

The site is going to have two main type of users. The first type of user is a user who visits the site to view the products and services rendered and possibly make a purchase. The second type of user is a registered member of the club who has authorized credentials to log into the chat forum and subscribe to other services. This type of user can do anything a regular user has access to plus additional functionalities.
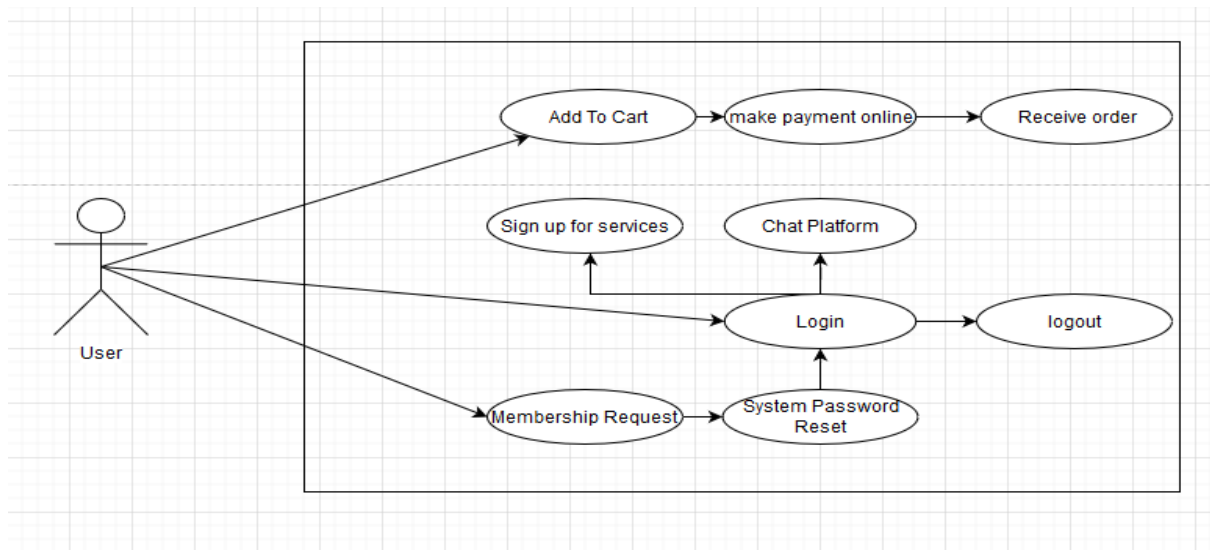
Figure 7.1 showing use case diagram for users

Table 2.2 showing use case table of users

Use Case: Membership request	Actor: user Requirement ID: US001
Stakeholder	Users
Description	This is to demonstrate club membership request.
Goal	User should be able to make a registration request.
Success Measurement	User successfully makes a registration request and an email sent to user to notify him or her of his or her registration.
precondition	User should be on the web application and click on the membership registration link.
Trigger	The application can only be accessed via the web through a browser.
Relationship	Upon a successful registration, user is shown success message and redirected to homepage.
Event Flow	The user clicks on registration link. 1. Registration page is displayed. 2. User enters registration details in the required fields. 3. Success message is shown, and user redirected. 4. User receives email to notify success of registration.
Use Case: Password reset and login	Actor: User Requirement ID: US002
Stakeholder	User
Description	This is to demonstrate membership password reset and log in of the system.
Goal	A member is successful in password reset and able to log in.
Success Measurement	Successful reset of password, correct credentials are accepted by the system and access given whilst improper credentials are rejected by the system.
precondition	A password reset link is sent to user
Trigger	The application can only be accessed via the web through a browser.
Relationship	When the user is successful with the password reset, the user can log in.
Event Flow	The user clicks on the password reset link sent into the email. 1. Password reset page is displayed.

	2. User sets password and confirms it. 3. When successful, the user is redirected to the chat platform and can also subscribe to other services
Use case: product purchase	Actor: User Requirement ID: US003
Stakeholder	User
Description	This is to demonstrate the process a user goes through to make purchase of products on the e-commerce platform
Goal	A user should be able to make purchase on the platform and make online payment
Success Measurement	Successful purchase made on the platform.
precondition	User should be on the web application.
Trigger	The application can only be accessed via the web through a browser.
Relationship	The user visits the e-commerce platform, adds to the cart, proceeds to checkout and finally proceeds to make payment or place purchase request.
Event Flow	1. The user clicks on a product to add to cart. 2. Proceeds to checkout. 3. Makes payment or purchase request. 4. Tracks order until it is received
Use case: Service subscription	Actor: User Requirement ID: US003
Stakeholder	User
Description	This is to demonstrate the process a user goes through to subscribe for services offered by the club.
Goal	A user should successfully subscribe for services
Success Measurement	Successful subscription made by members.
precondition	Only registered members of the club and should be logged in.
Trigger	The application can only be accessed via the web through a browser.
Relationship	The user visits the application, selects the service to subscribe to, prompted to log in if not logged in already.

Event Flow	1. The user selects the service to subscribe to.
	2. Prompted to log in if user is not logged in already.
	3. Fills in details and submits subscription request

2.4   Operating Environment

The application will be deployed on an Apache HTTP Server with PHP and MySQL database installed.

2.5   Functional Requirements

This part will be focusing on the key functional requirements of the application

2.5.1   Make purchase/subscribe to a service

A user of the application should be able to make a purchase of merchandise and subscribe for services.

User Requirement: the user visits the web application and selects the products to purchase, adds them to cart, proceed to checkout and make payment online. The user selects the services they want to subscribe to and complete the subscription process.

System Requirement: the web application should support the user purchase journey online and support user subscription to a service.

Input/Output: User selects a product to add to cart. Confirms online payment details. The user clicks to subscribe to a service, fills in the details and submit.

2.5.2 User Management

Admin should be able to accept request from members requesting to be part of the club, give access to the secure chat platform, ship merchandise orders made by customers

User Requirement: administrator should log into the dashboard and perform the necessary actions.

System Requirement: the system should assist the administrator in managing users of the application from the dashboard.

Input/Output: Administrator enters correct log in credentials and the systems allows log in. The administrator will certify user details and grant access to users.

2.5.3 Add/Update Products and Services

Administrator should be able to add and update products, track orders, ship orders, confirm purchases made. Administrator should approve request for services.

User Requirement: administrator should log into the administrator dashboard, from there the necessary actions can be made.

System Requirement: The system should allow the administrator to log in and be able to carry out the necessary transactions.

Input/Output: the administrator keys in the required details of the product or service, uploads the required images and attachments. The products and services page get updated with the current updates.

2.5.4   Chat platform

Members of the club should be able to interact with each other through the secured chatting platform.

User Requirement: User should log into the system and navigate to the chat forum.

System Requirement: The application should allow for navigation into the chat forum and provide a seamless chat experience.

2.6   Non-Functional Requirements

2.6.1   Security

The application should be highly secured to avoid unauthorized access to the chat platform and the admin dashboard. The chat forum is solely for only registered members of the club hence they should be the only people who can join. The online payment portal should be secured to ensure that transactions are complete.

2.6.2   Performance

The performance of the system should be optimum with a lower latency rate. The application should be robust to accommodate multiple users at the same time. The chat portal should have a high availability rate with low latency. Members on the platform should be able to interact without any interference in the system.

Chapter 3: System Architecture

3. 1   Overview

The application is going to be web-based. The application will be hosted on a web server running Apache Tomcat with PHP installed and MySQL database. Two main architectural designs will be used. That is Client Server Architecture and Model View Controller Architecture (MVC). The two architectures are explained below.

3.2   Client Server Architecture

This architecture takes the pipe programming technique where we have a producer and a consumer. In this case, the producer is the server and the consumer is the client. A client is any device or another program which makes request of a resource from the server [5]. The server in this case houses all resources of the application. It houses the critical resources of the application to make it available to clients at all time and in high end. The server will as well house the database to be used for the project.
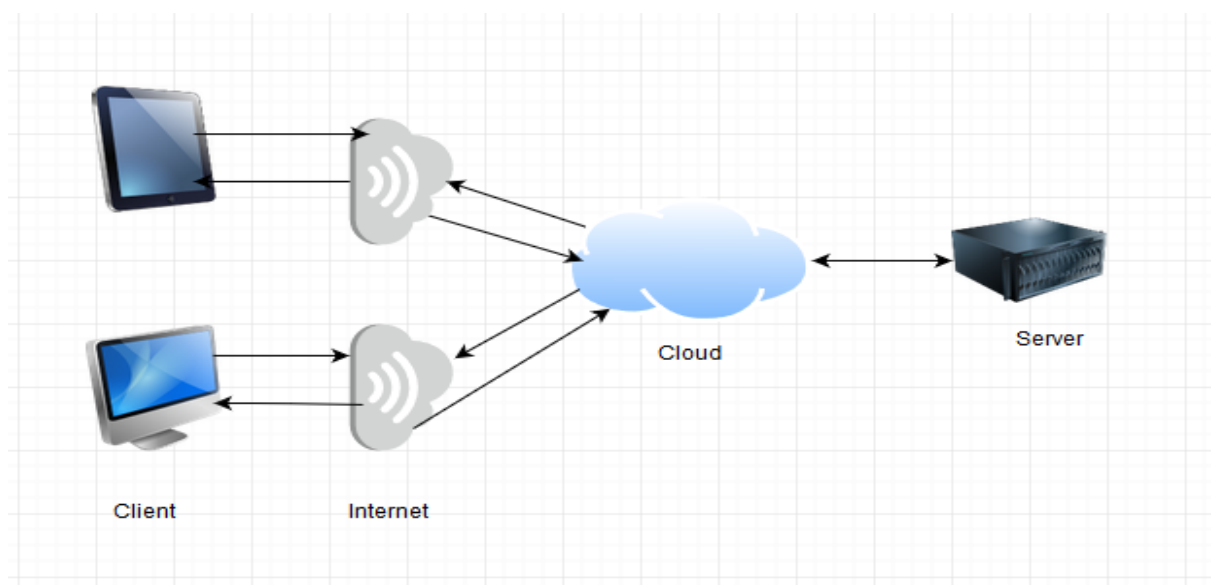
Figure 8.0 depicting client server architecture interaction.

3.3   Model View Controller (MVC)

The model view controller architecture allows for the separation of the different components of the application into layers or units called the model, view and the controller. Each unit performs a unique task at every point in time. The model component is responsible for designing and describing the various data models, the logics and relationship between entities. It handles the interactions that happen with the database and communicate with controller [6]. The view is the foreground and front facing part of the application which the user interacts with. It is the user interface written mainly in HTML, CSS and JavaScript. The view communicates with the controller to get updated dynamically. The controller is the middle entity which facilitates the interaction between the view and the model. The controller receives data or request from the view. It interacts with model to work on such request. It handles GET, POST and URL requests [6].
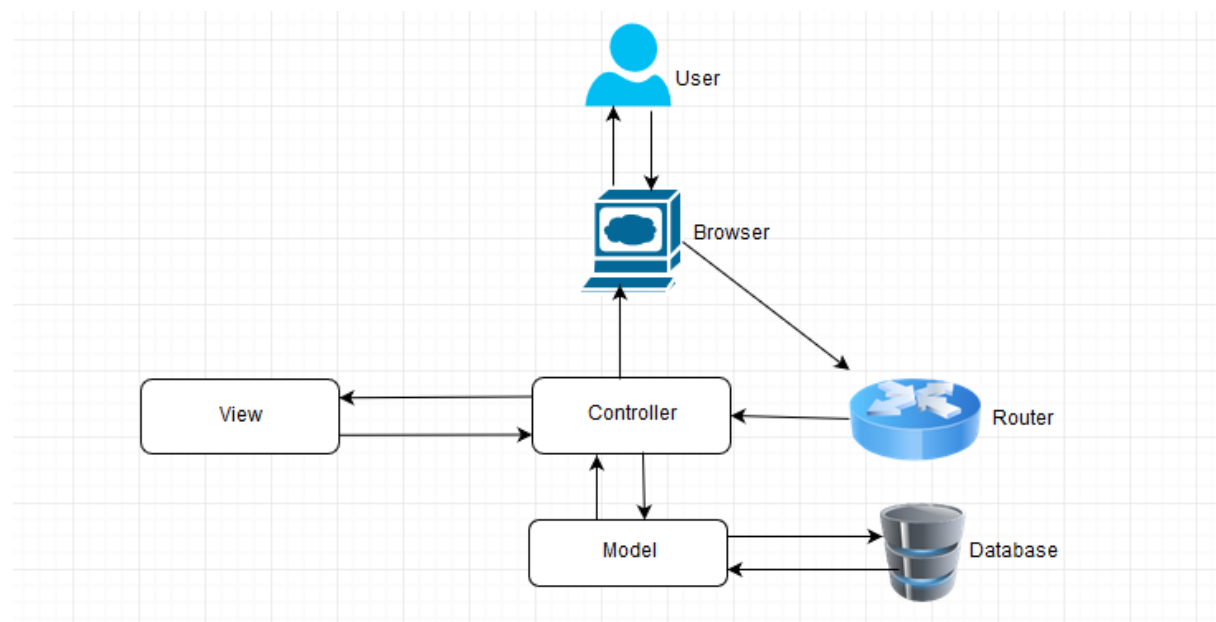
Figure 8.1 depicting the interaction of MVC Architecture.

From above, as the user tries to access the web application through a browser, it passes through the gateway router to the controller. The controller then understands the resource

being requested and makes a call to the appropriate view to be displayed. It also makes data request to the model upon request. The model communicates directly to the database.

3.4   **Interface Design**

Considering the nature of the application, it will have the main interface that will help in the navigation to the various sections of the application. There will as well be an administrator interface.

Application Homepage
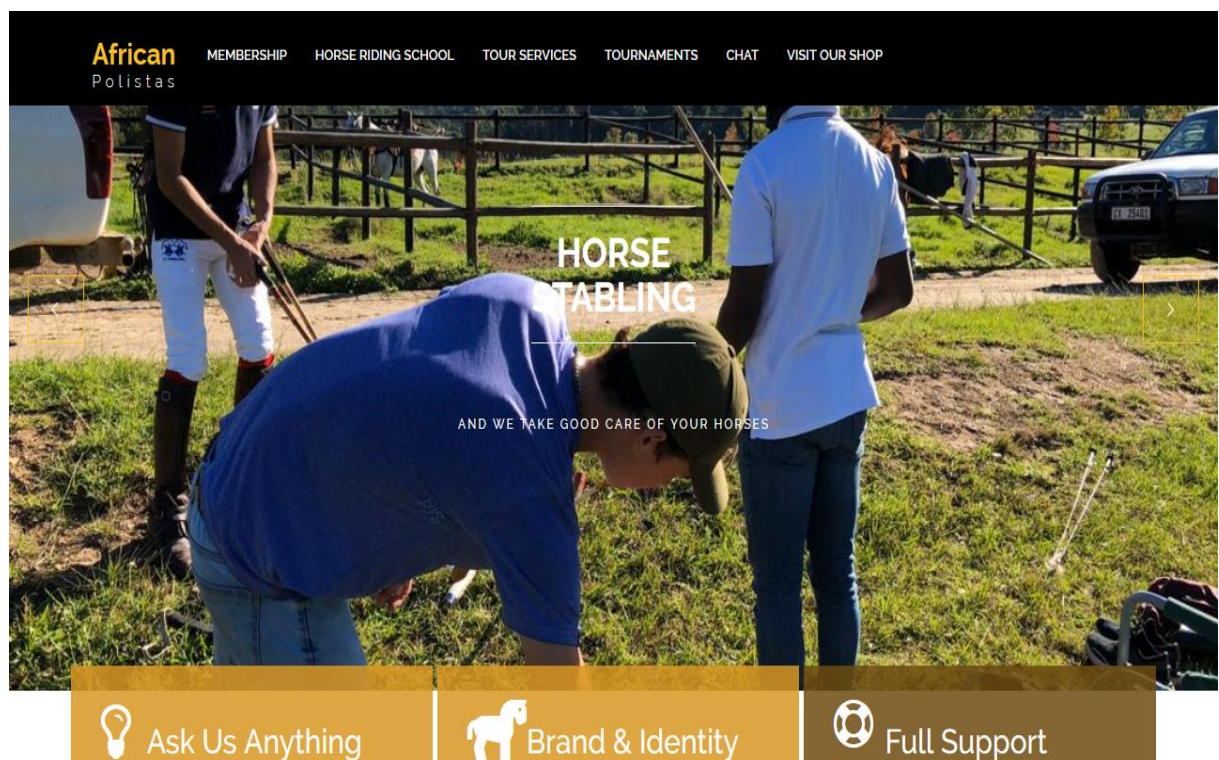
Figure 8.2 showing the landing page when the application is visited.

The landing page provides links that lead to the various sections of the application such as the e-commerce site, chat platform, horse riding school, membership registration among others. This same landing page has a chatbot to help answer user questions and help in application navigation.

E-commerce
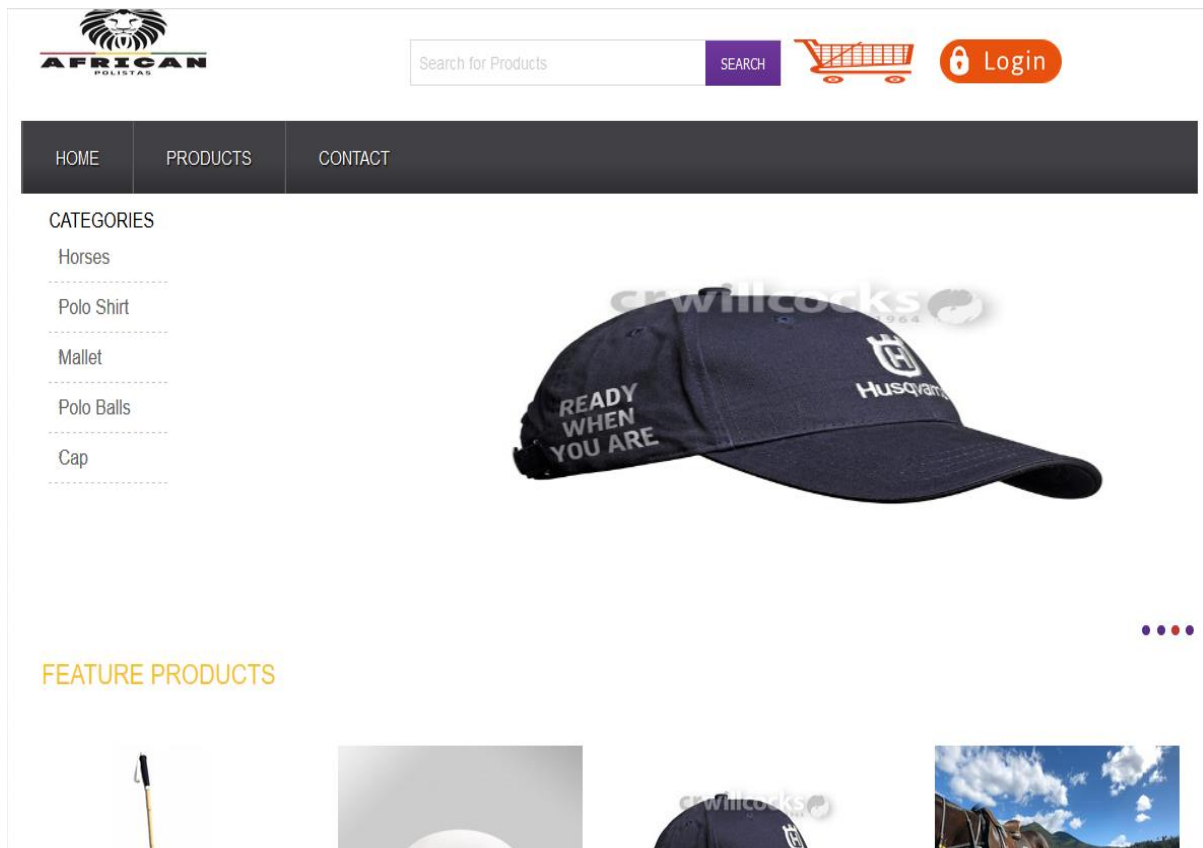
Figure 8.3 depicting the home page of the e-commerce platform.

Through this page, users can navigate all products and make purchases. Users can as well track their orders. Since the e-commerce site will be targeting only polo players and polo enthusiasts, hence the need to keep it simple and succinct.

Chat Platform
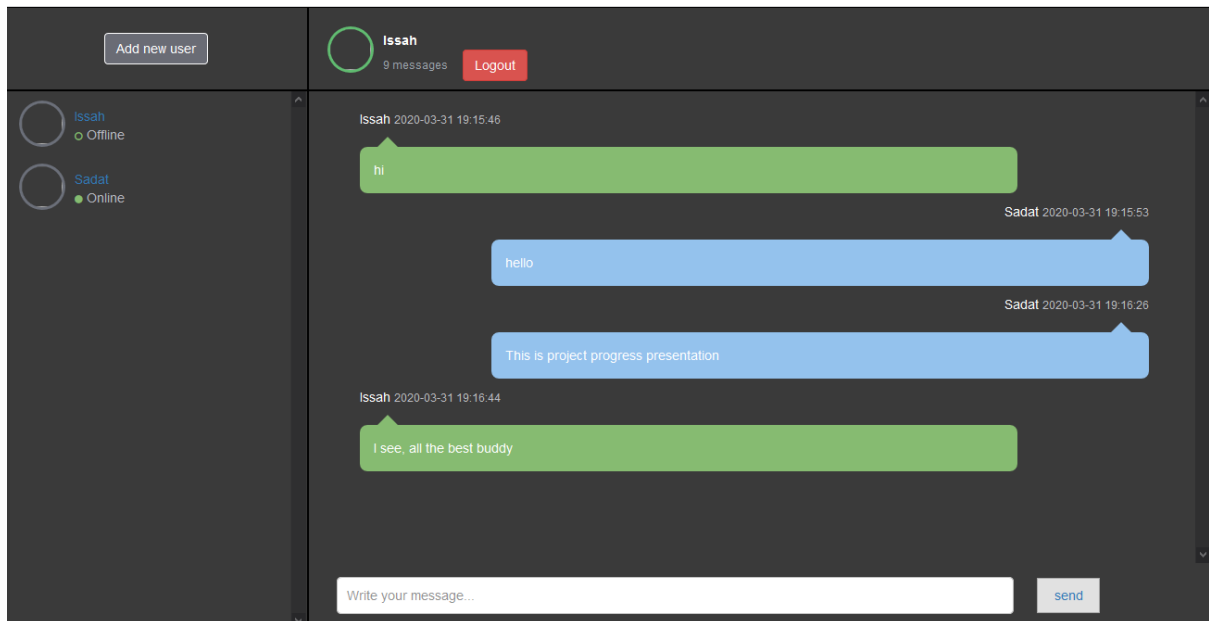
Figure 8.4 depicting home page of the chat platform

This chat platform is for only registered members of the club. The members of the club can interact with other members on the platform. This will serve as a great networking platform for members of the club in various places. Even though this a web interface, it appears very well on mobile devices or smaller screens

Administrator Dashboard
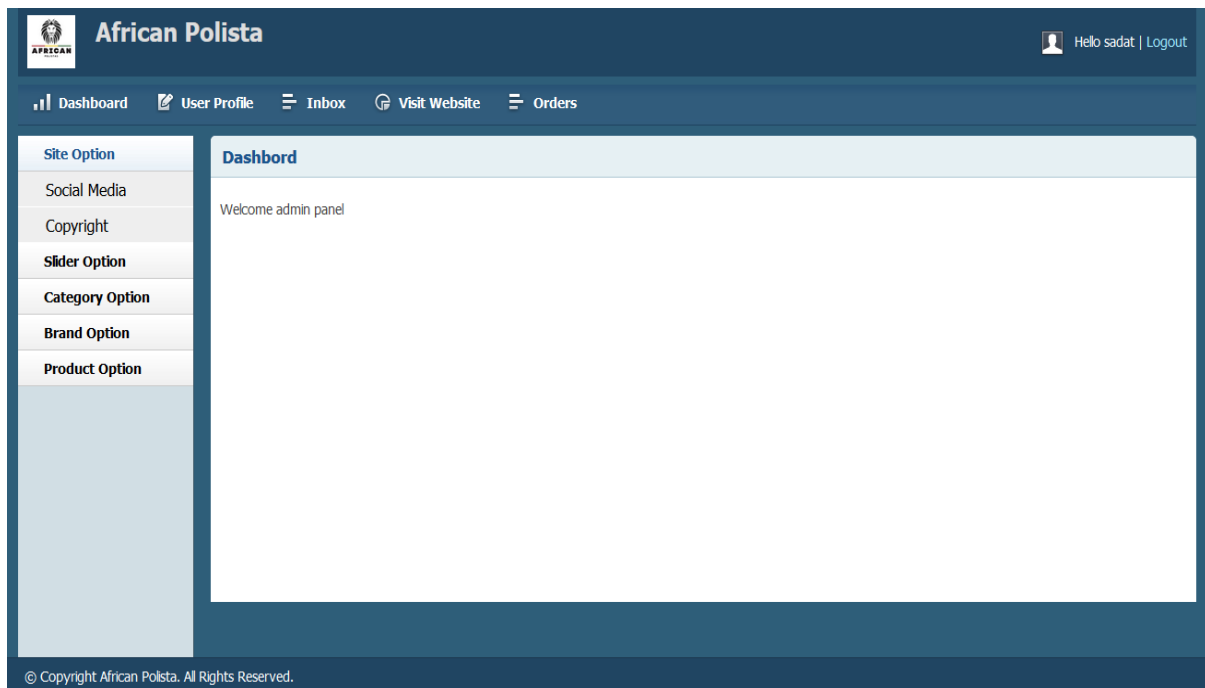
Figure 8.5 depicting administrator dashboard

With this dashboard, administrator can manage users, services, e-commerce activities and the secured chat platform. Various details of the sections of the dashboard will be explored at the implementation chapter.

3.5   Database Design

The primary database used in this project is MySQL database. It is open source, reliable, secure and easily compatible with many hosting service providers whilst providing easy management dashboard. I am maintaining a single database for the application. The ecommerce application has numerous tables such as; orders, products, users, cart, brands, categories, customers etc. The chat application also tables such as chat_msg, chat_users etc.

chat_msg
- ID INT(11)
- sender_name VARCHAR(50)
- receiver_name VARCHAR(50)
- msg_content VARCHAR(255)
- msg_status TEXT
- msg_date TIMESTAMP
- Indexes

shop_admin
- ID INT(11)
- Name VARCHAR(30)
- User VARCHAR(30)
- Email VARCHAR(30)
- Pass VARCHAR(50)
- Level TINYINT(4)
- 2 more...
- Indexes

shop_users
- userID INT(11)
- fname VARCHAR(255)
- lname VARCHAR(255)
- city TEXT
- zip VARCHAR(255)
- email VARCHAR(255)
- addr VARCHAR(255)
- country VARCHAR(255)
- phone VARCHAR(255)
- pass VARCHAR(32)
- Indexes

wishlist
- wishID INT(11)
- userID INT(11)
- productID INT(11)
- product_name VARCHAR(255)
- price FLOAT(10,2)
- img VARCHAR(255)
- Indexes

chat_users_has...
- chat_users_ID INT(11)
- chat_msg_ID INT(11)
- Indexes

chat_users
- ID INT(11)
- first_name VARCHAR(20)
- last_name VARCHAR(20)
- phone TINYINT(4)
- pass VARCHAR(255)
- email VARCHAR(50)
- profile VARCHAR(50)
- 4 more...
- Indexes

shop_product
- productID INT(11)
- product_name VARCHAR(2555)
- catID INT(11)
- brandID INT(11)
- descr TEXT
- price FLOAT(10,2)
- img VARCHAR(255)
- typ TINYINT(4)
- Indexes

shop_cart
- cartID INT(11)
- sessionID VARCHAR(255)
- productID INT(11)
- product_name VARCHAR(255)
- price FLOAT(10,2)
- qty INT(11)
- img VARCHAR(255)
- Indexes

pass_reset
- id INT(11)
- email VARCHAR(255)
- token VARCHAR(255)
- 1 more...
- Indexes

shop_brand
- brandID INT(11)
- brand_name VARCHAR(255)
- Indexes

shop_category
- catID INT(11)
- catName VARCHAR(255)
- Indexes

shop_orders
- orderID INT(11)
- userID INT(11)
- productID INT(11)
- product_name VARCHAR(255)
- qty INT(11)
- price FLOAT(10,2)
- img VARCHAR(255)
- status INT(11)
- time TIMESTAMP
- Indexes

Figure 8.6 showing entity relationship between the various tables in the database.

From the diagram above, it will be seen that, there exist a varied relationship between the various entities. For example, the relationship between shop_users and shop_orders is one to many. This is because we can have one user with many orders.

Chapter 4: Implementation and Testing

Description of the implementation and component testing will be done concurrently. That is test results will be shown side by side implementation results.

4.1   Tools and Libraries

HTML5: Hypertext Markup Language (HTML) is used in the project to define the structure of the various web pages. HTML elements tell the browser how to display the contents of the application. HTML5 is the latest version of the HTML standards [8]. This new standard comes with a lot of new features and functionalities improving structure and performance [8]. Among the great improvements are semantics, multimedia, offline and storage among others.

CASCADING STYLE SHEET (CSS): This document is used to specify the style, colours, layouts, and other aesthetic designs. CSS adds style to HTML elements or tags. As explained above, HTML provides the structure of the web application and CSS adds aesthetic styles to it [9].

JAVASCRIPT (AJAX, jQuery): JavaScript is a programming language, or a scripting language used to design dynamic web pages that dynamically update contents, create animations etc [10]. jQuery Ajax was used in the application to build the interactive chat platform where users can send and receive seamless messages. This same technology was used to implement the various third-party APIs in the project.

PHP: PHP is a popular server-side scripting language used in this project as the main backend language. PHP is very useful in the project especially in database interactions [11]. The application relied heavily on database interaction and this language was very helpful.

MySQL: MySQL relational database was used in the project. It is open-source, reliable, secure and easily compatible with many hosting service providers whilst providing easy management dashboard [12].

THIRD PARTY APIs: The third-party APIs used in the project are Deywuru SMS API from Npontu Technologies (This API is to facilitate the sending of SMS updates to members of the club), Flutterwave payment API (This API is to facilitate online payment for goods purchased).

APACHE SERVER (XAMPP): XAMPP is an open-source cross-platform tool used for web services and contains apache server, MariaDB, PHP, Perl programming; providing a localhost server functionality [13].

4.2   Programming Paradigm

The project was built using object-oriented approach of programming. This approach was adopted because similar codes will be required at different sections of the project. This was going to allow for code reusability and scalability. I have implemented this using PHP classes. With PHP classes, I can dynamically reuse the same code in different sections of the project.

```php
<?php require (dirname(__FILE__).'../config/config.php');?>

<?php
 class Database {

    public $host = DB_HOST;
    public $user = DB_USER;
    public $pass = DB_PASS;
    public $dbname = DB_NAME;

    public $link;
    public $error;


    public function __construct(){
        $this->connectDB();
    }
```

```php
    /**
     * database connection
     *
     * @return void
     */
    private function connectDB(){
        $this->link = new mysqli($this->host,$this->user, $this->pass, $this-
            >dbname);
        if(!$this->link) {
            $this->error ="connection fail".$this->link->connect_error;
            return false;
        }
    }


/**
 * function to accept sql select statement
 * @param [sql] $query
 * @return void
 */
public function select ($query){
  $result = $this->link->query($query) or die ($this->link->error.__LINE__);
  if($result->num_rows > 0){
    return $result;
  } else {
    return false;
        }
    }



/**
 * function to accept sql insert statement
 *
 * @param [type] $query
 * @return void
 */
public function insert($query){
    $insert_row = $this->link->query($query) or die ($this->link-
        >error.__LINE__);
    if($insert_row){
      return $insert_row;
      exit();
     } else {
       return false;
        }
    }
```

Figure 9.0 showing code snippet of database class.

The above code snippet shows a database class with methods. For example, the function select($query) and insert($query) can used in any section of the project which requires database interaction.

Another example of PHP class implementation is sessions class. See below.

```php
<?php

/**
*session class
 */

class Session{

    public static function init(){
        session_start();
    }

    /**
     * set session variables
     *
     * @param [session key] $key
     * @param [session value] $val
     * @return void
     */
    public static function set($key, $val){
        $_SESSION[$key] = $val;
    }

    /**
     * get session value by applying session key
     *
     * @param [session key] $key
     * @return void
     */
    public static function get($key){
        if(isset($_SESSION[$key])){
            return $_SESSION[$key];
        }
        else{
            return false;
        }
    }
```

Figure 9.1 depicting session class.

The application will make use of sessions to manage users on the platform. Different sections of the application will use sessions hence it is prudent I maintain a PHP class to enable the identification of users. Especially on the e-commerce platform and the chat platform.

E-COMMERCE PLATFORM

4.3   Admin Dashboard

Administrator Login Page
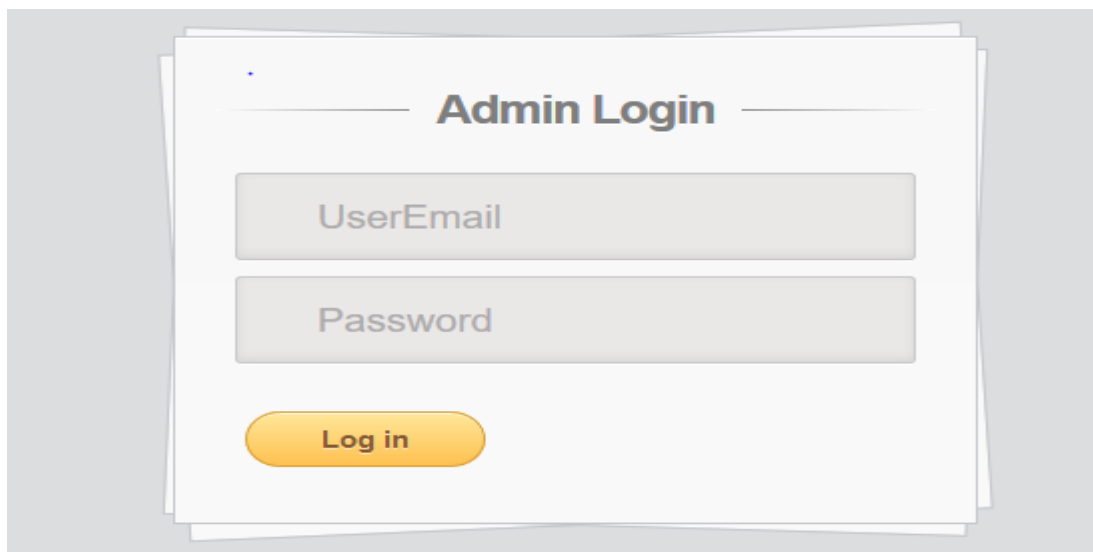
Figure 9.2 showing admin login page

Administrator who is granted access can use this portal to gain access to the main administrator dashboard.

An administrator entering a wrong login credential
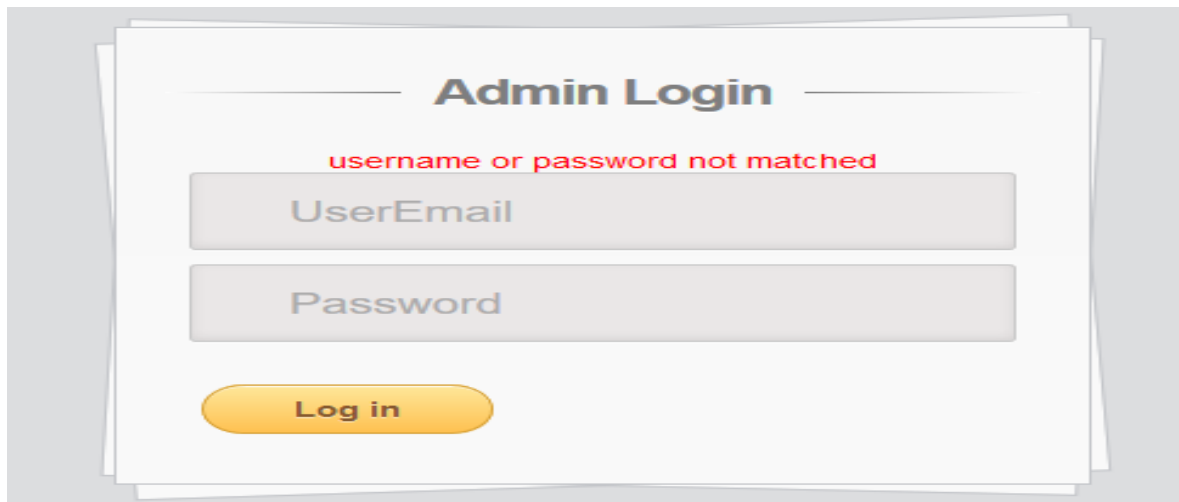
Figure 9.3 showing login fail for admin with wrong credentials

When an administrator enters a wrong login credentials, a notification is shown to alert the admin about the use of wrong credentials.

4.3.1 Admin Dashboard

The interface when administrator successfully logs in.
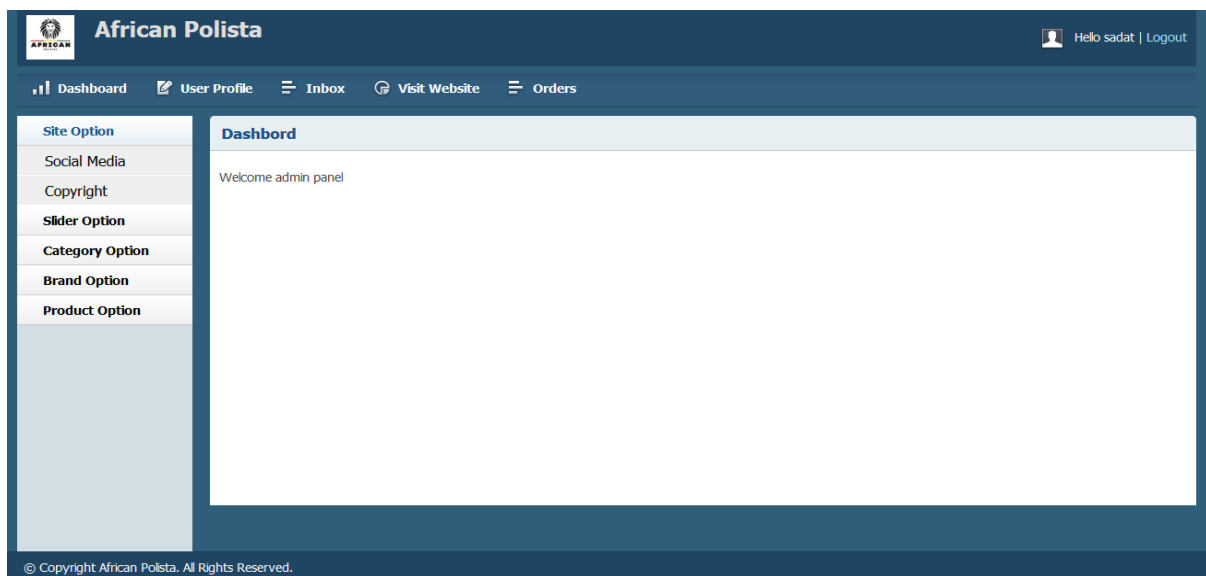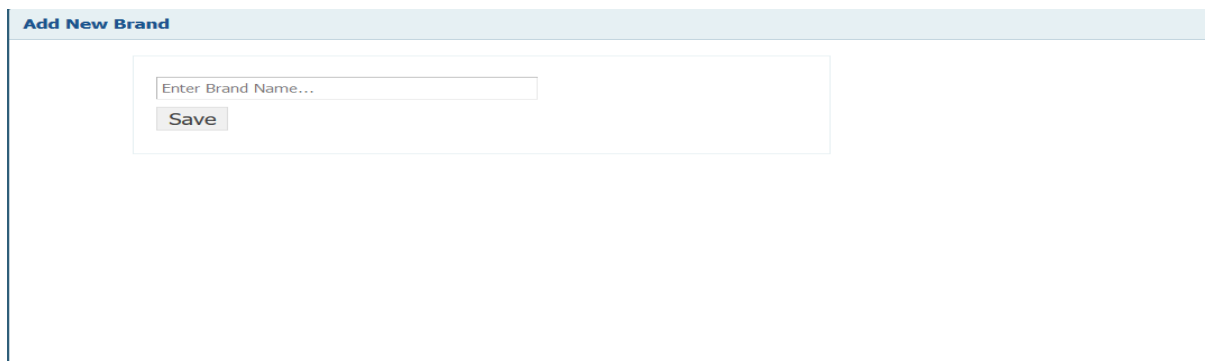
Figure 9.4 depicting the administrator dashboard.

4.3.2 Add Product Brand

This functionality will enable the administrator to add new brands to the already existing brands in the dashboard.
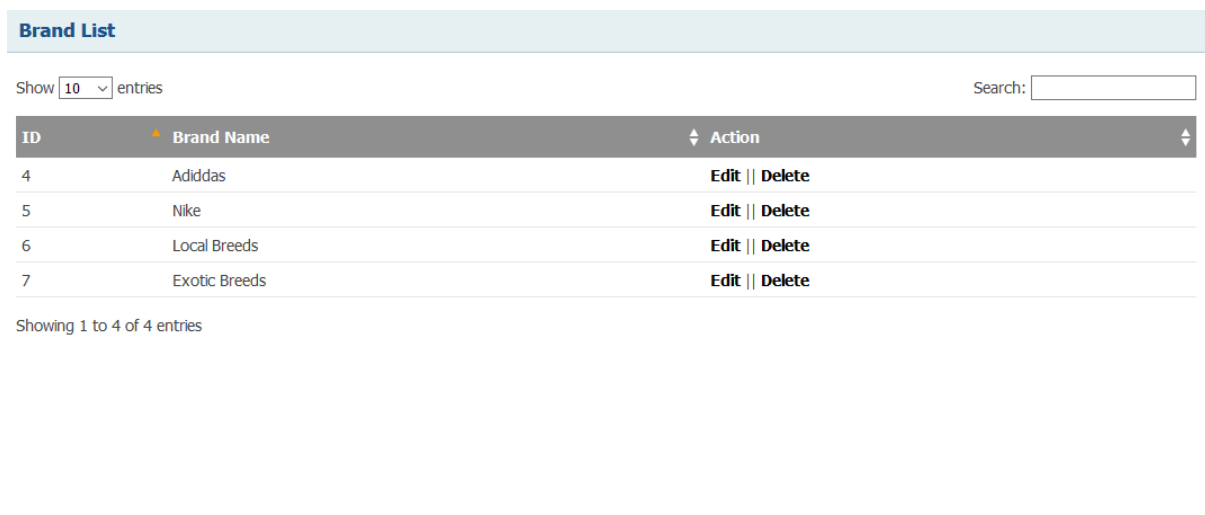
Figure 9.5 depicting add product brand page in the dashboard

4.3.3 Brand List

This page displays the brand list added to the database. The administrator from this page can edit a brand and delete same if the need be.

Figure 9.6 depicting brand list page.

4.3.4 Add Category

This page is to enable administrator to add new category item to the database.

Enter Category Name...

Save

Figure 9.7 depicting add category page

4.3.5 Product Category List

This page is to list all the product categories available in the database. Aside from listing the categories, the administrator can make edits to each category and delete if the need be. There is also a search functionality to aid the administrator in scrolling through a long list.

Category List

Show 10 ✔ entries   Search:
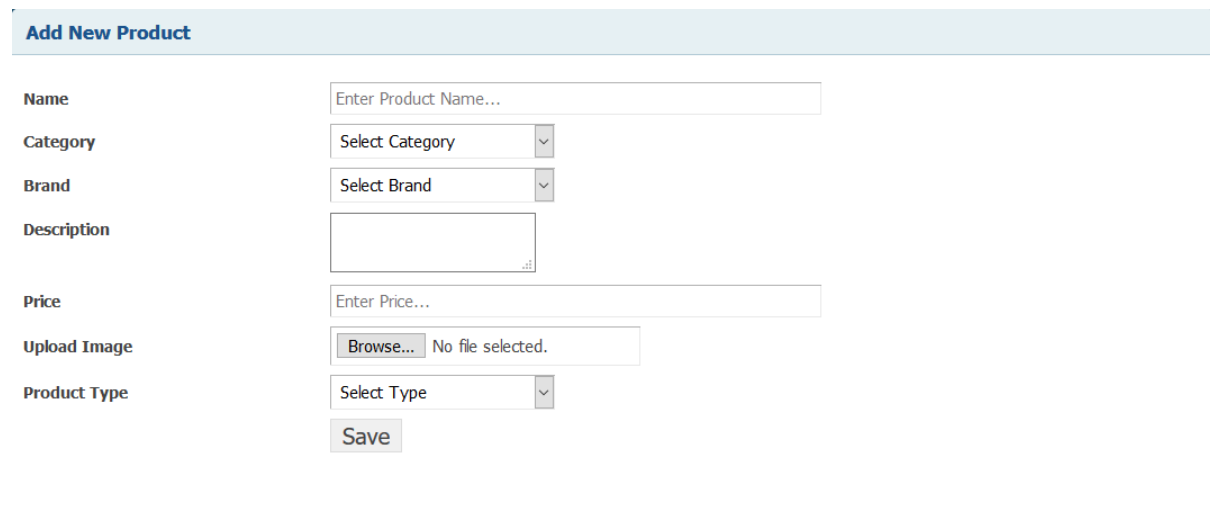
ID	Category Name	Action
8	Horses	Edit || Delete
9	Polo Shirt	Edit || Delete
10	Mallet	Edit || Delete
11	Polo Balls	Edit || Delete
13	Cap	Edit || Delete

Showing 1 to 5 of 5 entries

Figure 9.8 depicting category list

4.3.6 Add Product Page

This page will enable the administrator to add new products to the e-commerce site. Here the administrator key in the required details and uploads the appropriate image. The administrator also gets to select the brand and category which a product belongs to. Brand and Category have already been created; this will make it easy for displaying of products under their respective brands and categories.

Figure 9.9 depicting a page for adding new product

4.3.7 **Product List**

This page displays all products added by the administrator. Products here are the products customers can purchase. Administrator can make changes to the product details and delete any product if it is not needed again.

Products

Show 10 entries   Search: [   ]

ID	Product Name	Category	Brand	Description	Price	Image	Type	Action
7	Kijaro	Horses	Local Breeds	A very Good horse	30000.00		Featured	Edit || Delete
8	Pinnato	Horses	Exotic Breeds	Very Wild	400000.00		Featured	Edit || Delete
9	Sleek Shirt	Polo Shirt	Adiddas	Nice	76.00		General	Edit || Delete
11	Ball Zee	Mallet	Nike	Ball which doesnt fail	20.00		Featured	Edit || Delete
12	Magic Stick	Mallet	Adiddas	Stick which is accurate	34.00		Featured	Edit || Delete

Showing 1 to 5 of 5 entries

Figure 9.1.1 depicting products list page.

4.3.8 Orders Page

This page will enable the administrator to view all orders made. The page contains all the relevant information the administrator may need. If the product is shipped to the customer, the administrator can make changes by clicking on the action column beside the product. Confirmation of online payment is done on the dashboard of the payment API I am using.

Customer Order

Show 10 entries   Search: [   ]

Order ID	Order Date	Product	quantity	Price	Customer ID	Address	Action
1	2020-03-31 18:56:03	Magic Stick	1	34.00	4	View Address	Shipped
2	2020-04-15 17:44:05	Ball Zee	1	20.00	4	View Address	Shipped

Showing 1 to 2 of 2 entries

Figure 9.1.2 showing order table in the administrator dashboard

4.3.9   Flutterwave Dashboard

Flutterwave is the payment API I am using for the e-commerce platform. From this dashboard, the administrator can confirm payment made by customers. This payment API allows for making payment using mobile money regardless of the country, it allows for making payment using card; Visa and MasterCard.
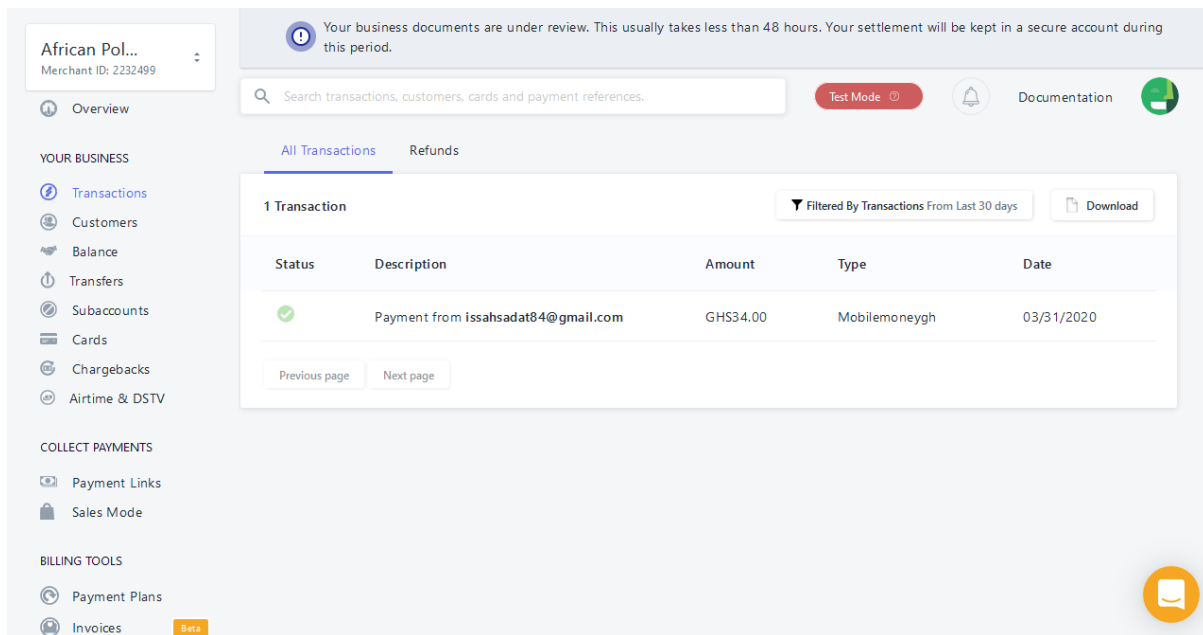
Figure 9.1.3 showing payment API dashboard.

4.4   E-commerce Customer Page

4.4.1   Homepage

This is the first homepage the user is presented with upon visiting the homepage. From this page the user can browse through by category or search products using the search button. The user can also decide to browse through the list of products available which are categorized as featured or non-featured.
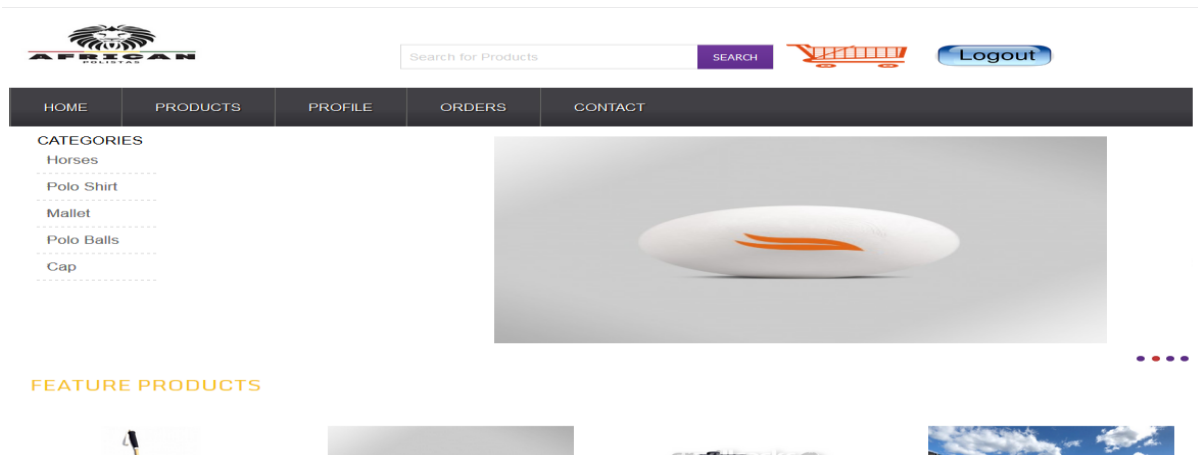
Figure 9.1.4: showing e-commerce site home page

4.4.2 Single Product View

This page displays if a user clicks on a product. The page has all information the user may need about the product. It has brand and category information. The user can see a brief description of the product and the price. The user can update the required quantity to be purchased. If the user is satisfied with the product, the user proceeds to add to the cart.
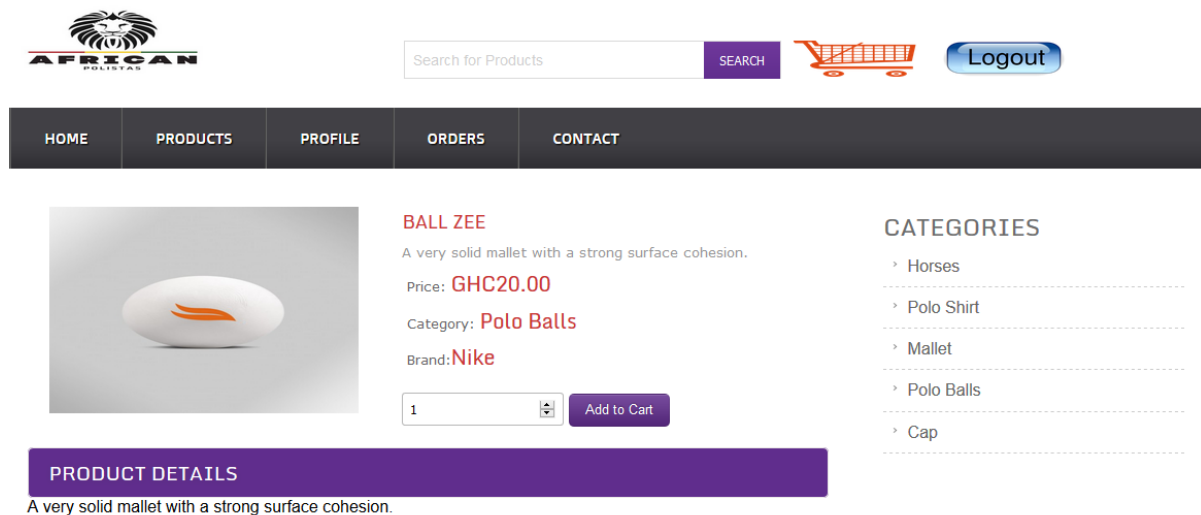
Figure 9.1.5 showing a single product page.

4.4.3 Cart Page

In the customer cart, users can update product quantity and delete a product. If the user feels satisfied with the cart items, the user can either proceed to checkout or continue shopping. If user decides to proceed to checkout, a log in is required.
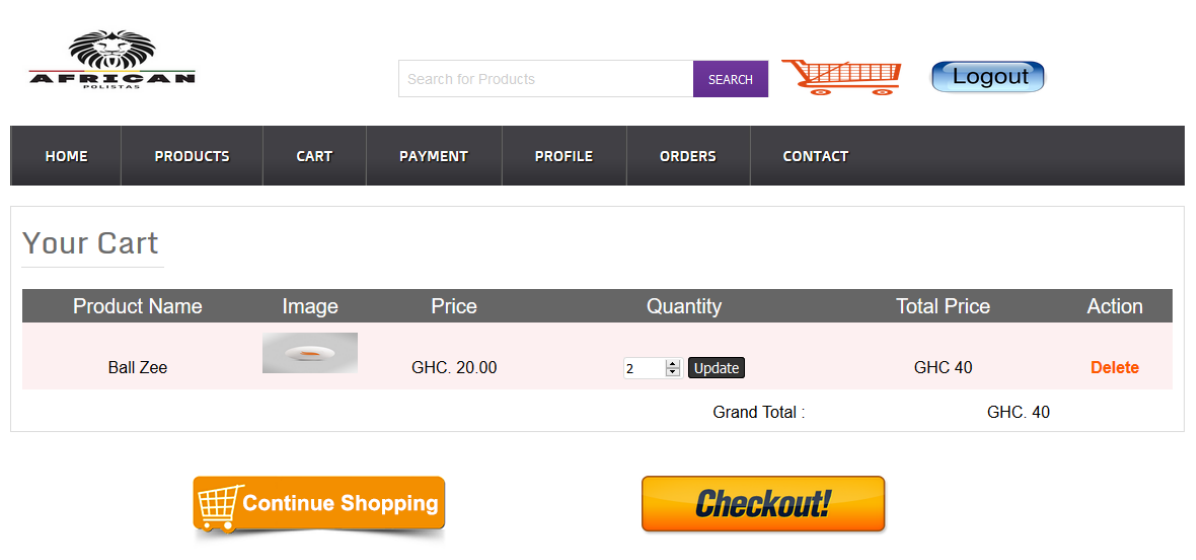
Figure 9.1.6 showing cart page

4.4.4 Sign in and Log in Page

On this page, already existing users can just log in, new users will have to register and log in. If an email is already registered into the system, a second registration will be declined by the system.

Test result when already registered email is used in a new registration
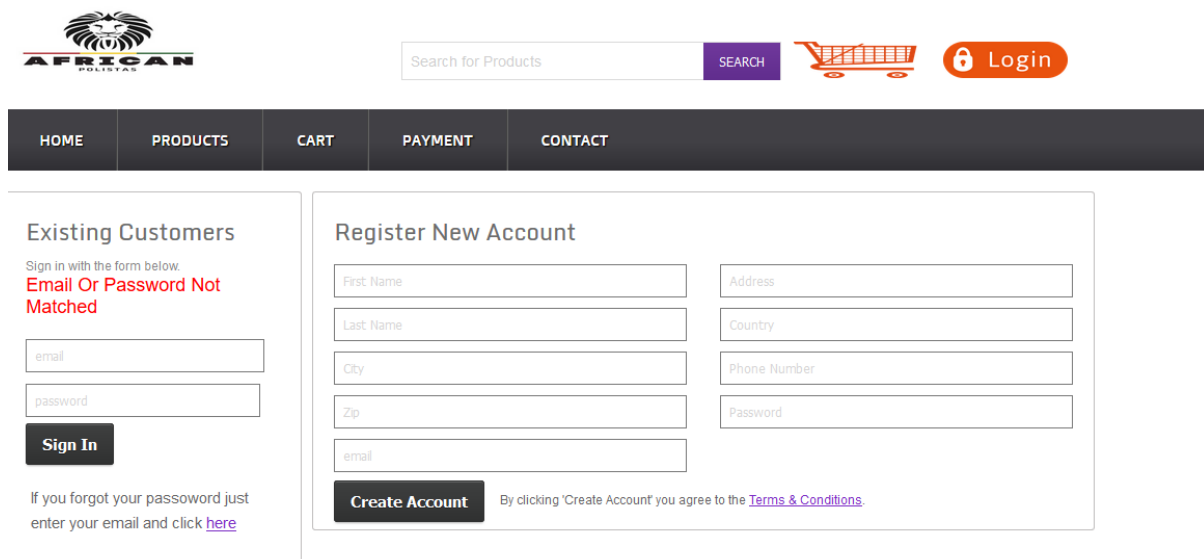
Figure 9.1.8 showing test result of duplicate email registration.

Test result showing when a wrong log in credential are used
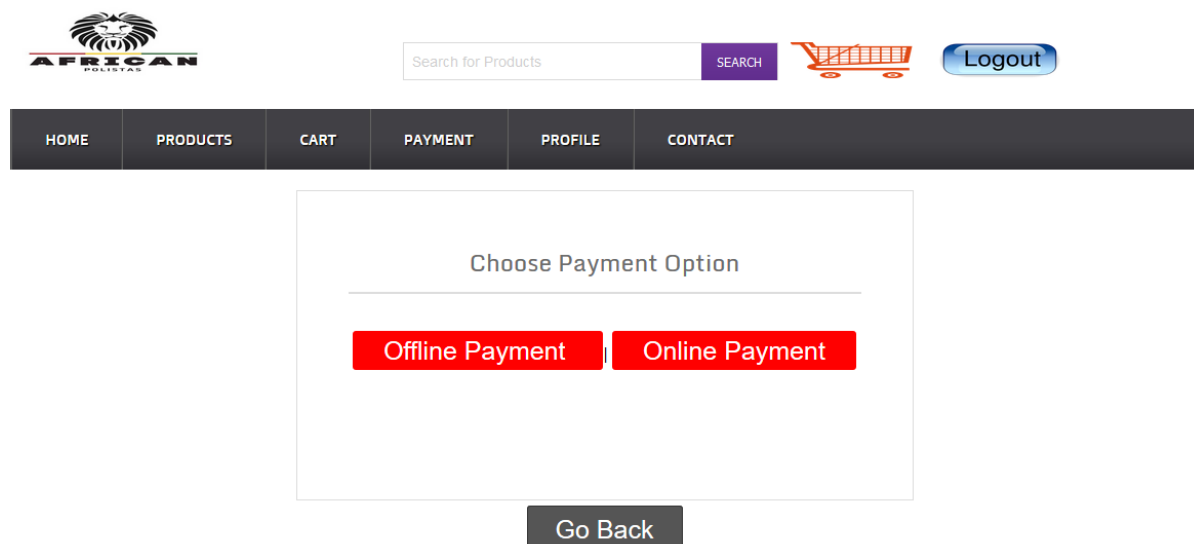
Figure 9.1.9 invalid log in credentials

4.4.5 Payment Option

After a user logs in, the user is then required to choose the best suitable payment option. For example, the purchase of a horse will require a user to choose offline payment because the user will have to do a physical inspection of the horse before proceeding to make a purchase.

Figure 9.2.0 showing payment option

4.4.6 Online Payment

This page displays the final purchase details of a customer. The customer confirms the products and the total amount. The customer can make changes to their details if any of their details is incorrect using the update button at your profile details.
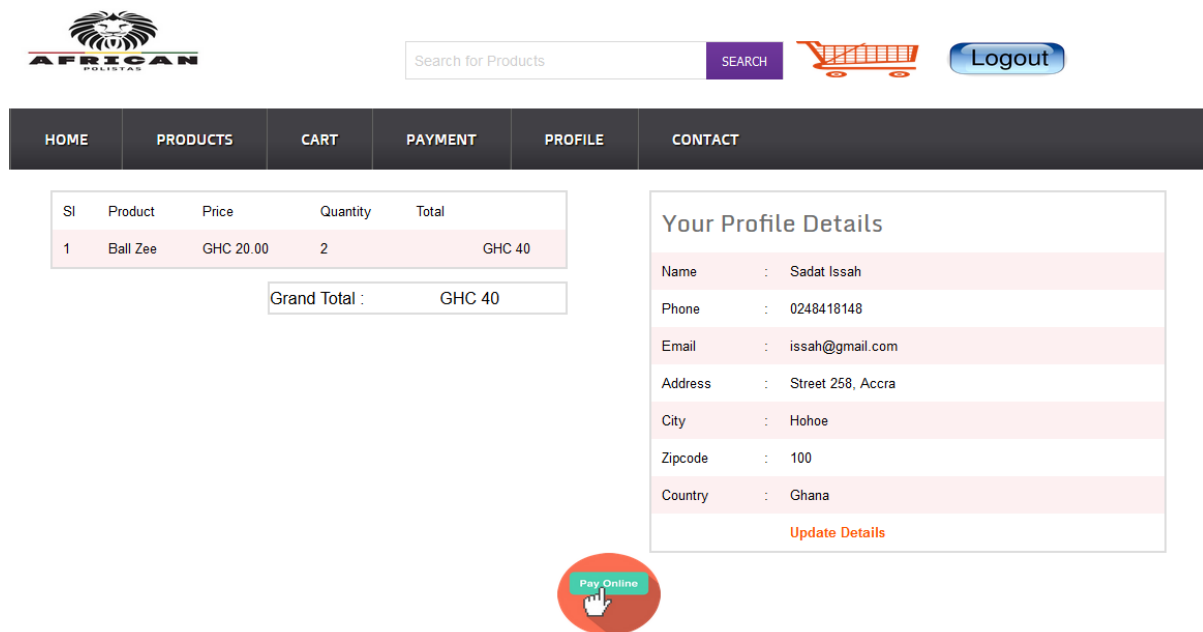
Figure 9.2.1 confirming purchase details

4.4.7   Payment Processing

This interface is provided by Flutterwave payment API. Here the buyer pays using a suitable medium, if mobile money, the user selects the network to make payments to. Also, the user can pay using card.
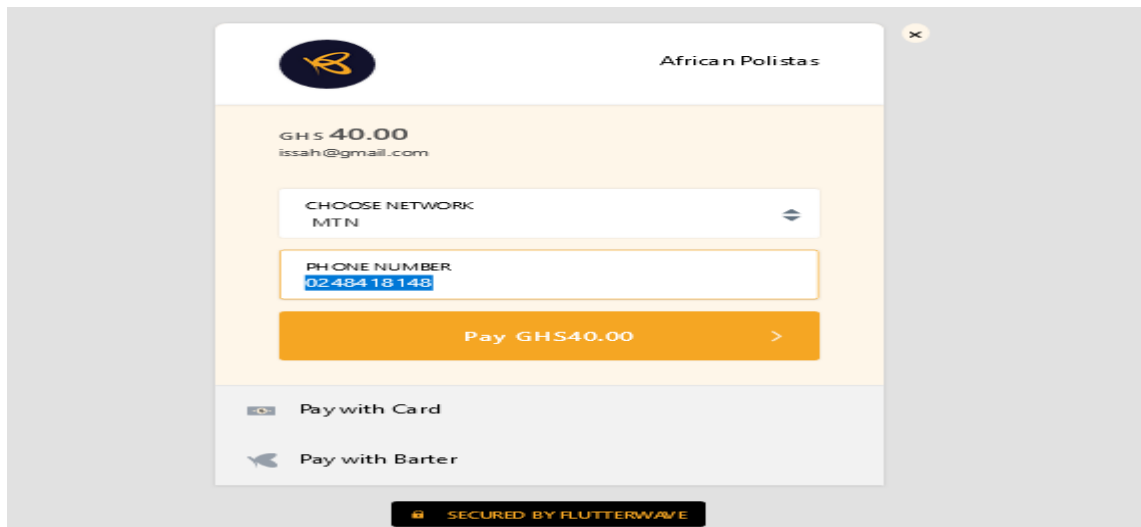
Figure 9.2.2 showing purchase confirmation

4.4.8 Customer Orders

In this page, the user will see all the ordered products. Here the user can track their status if the product is still pending or it's being shipped.

Figure 9.2.3 showing customer order page

4.4.9 Payment Notice

Payment notification sent to administrator as well as the user via mail. I am unable to get a payment notice for a customer because I am using testing credentials for the payment API, so actual payment is not done.

Figure 9.2.4 showing payment notification sent to the user

4.5   Chat Platform

The chat platform feature of the application is only accessible to registered members of the club. This will help connect members of the club across the world. On this chat application, members can have seamless interaction with all members of the club. There is a search area to be used to search for prescribed members. Members can change their details and profile pictures. Before a member gets access to the portal, they first must send a membership registration request. Administrator will then review the request, when the membership is accepted, an email is sent to the user with a link to reset his/her password to get access to the system.

4.5.1 Membership4 Registration

Here, a prospective member fills in his or her details. When this is done and submitted, the user gets an email acknowledging receipt of the registration request.

Figure 10.1.1: showing membership registration

4.5.2 **Registration Notification**

Upon registration, an email is sent to user to acknowledge receipt.
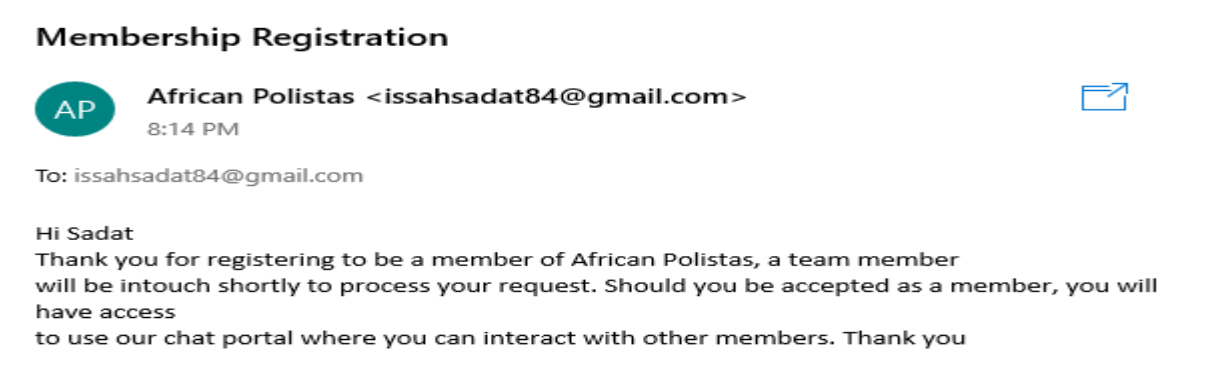
Figure 10.1.2: showing email notification of registration

4.5.3 **User Profile on Administrator Dashboard**

This page allows administrator to approve or disproves requests to be a member. When a request is verified and accepted, the 'Password Set' button is clicked, this will send an email to the user containing a password reset link.

Figure 10.1.3: showing user profile in the administrator dashboard.

4.5.4 Password Reset Email

This email is to notify the user that, he/she is accepted as a member of the club. It contains a link which the user can use to reset his/her membership password. This privilege will allow the member to subscribe to services the club offers including the chat platform. This email is not in the best format because I have been unsuccessful in getting a prescribed emailing format of the club.

Figure 10.1.4 depicting password reset notice

4.5.5 Password Reset Form

Form asking user to enter his/her password and confirm same. When an inconsistent password is entered, user is prompted to re-enter the passwords.

Figure 10.1.5: showing password reset form

4.5.6 Chat Platform Sign In
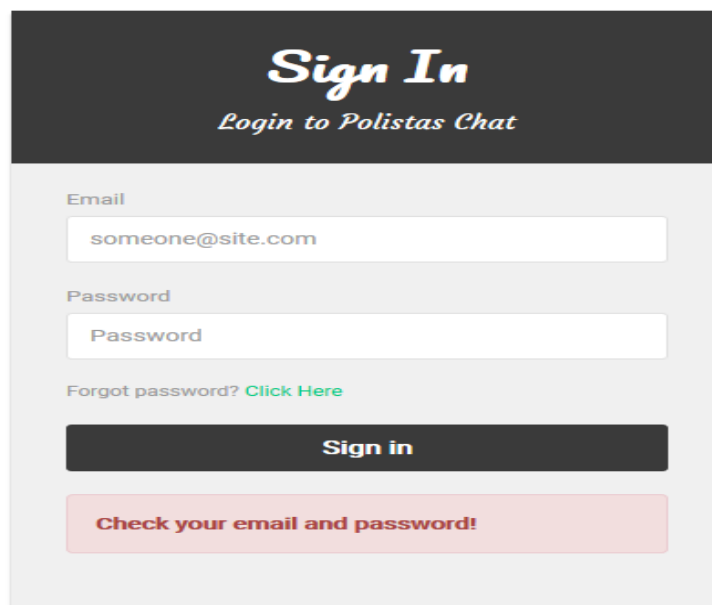
This page allows members who want to sign into the chat platform to do so. The form accepts the email used in membership registration and the password the user created. The form has a forgot password link which enables a registered user to request a password reset. When the link is clicked, a new page pops up, the user is then required to enter a verified email, if the email is already registered, a password reset link is sent to the user's email.

Figure 10.1.6: showing chat platform sign in

Test Result showing when a wrong log in credential is entered.

*Figure 10.1.7: image showing wrong login credentials to the chat platform*

4.5.7 Profile Edit

This page is to allow a user to make changes to his/her profile details. For example, the person can change profile picture.
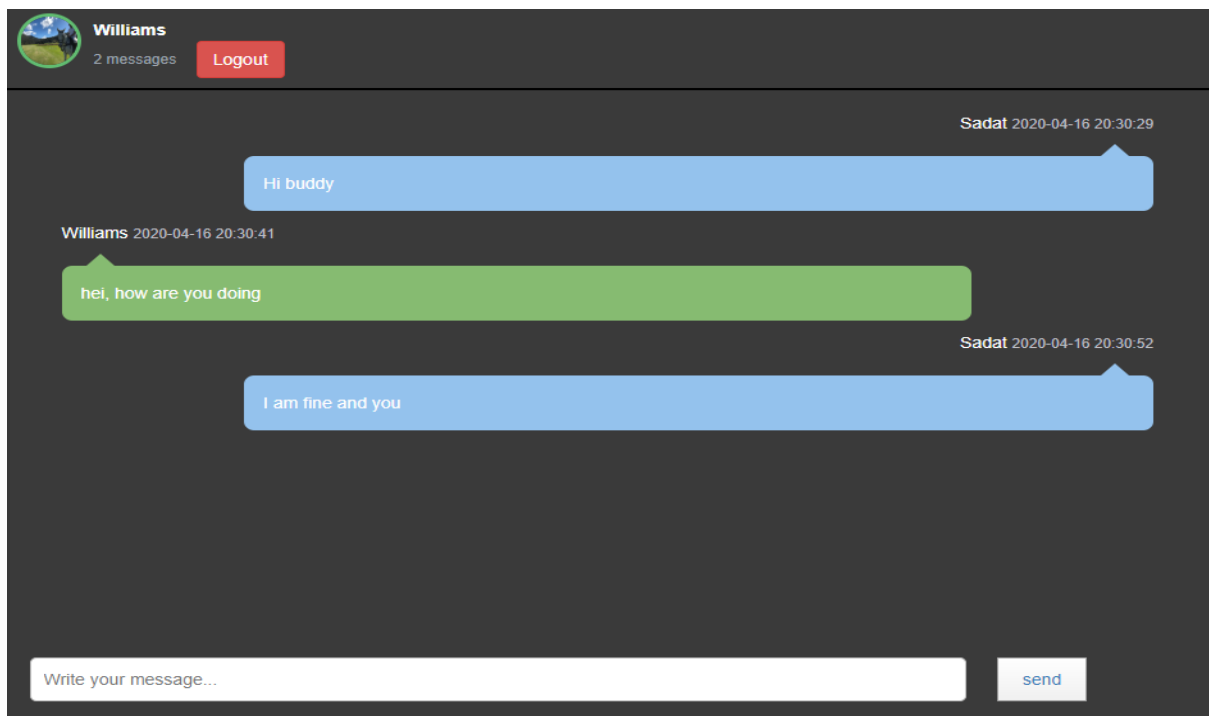

Figure 10.1.8: showing change account settings of a member
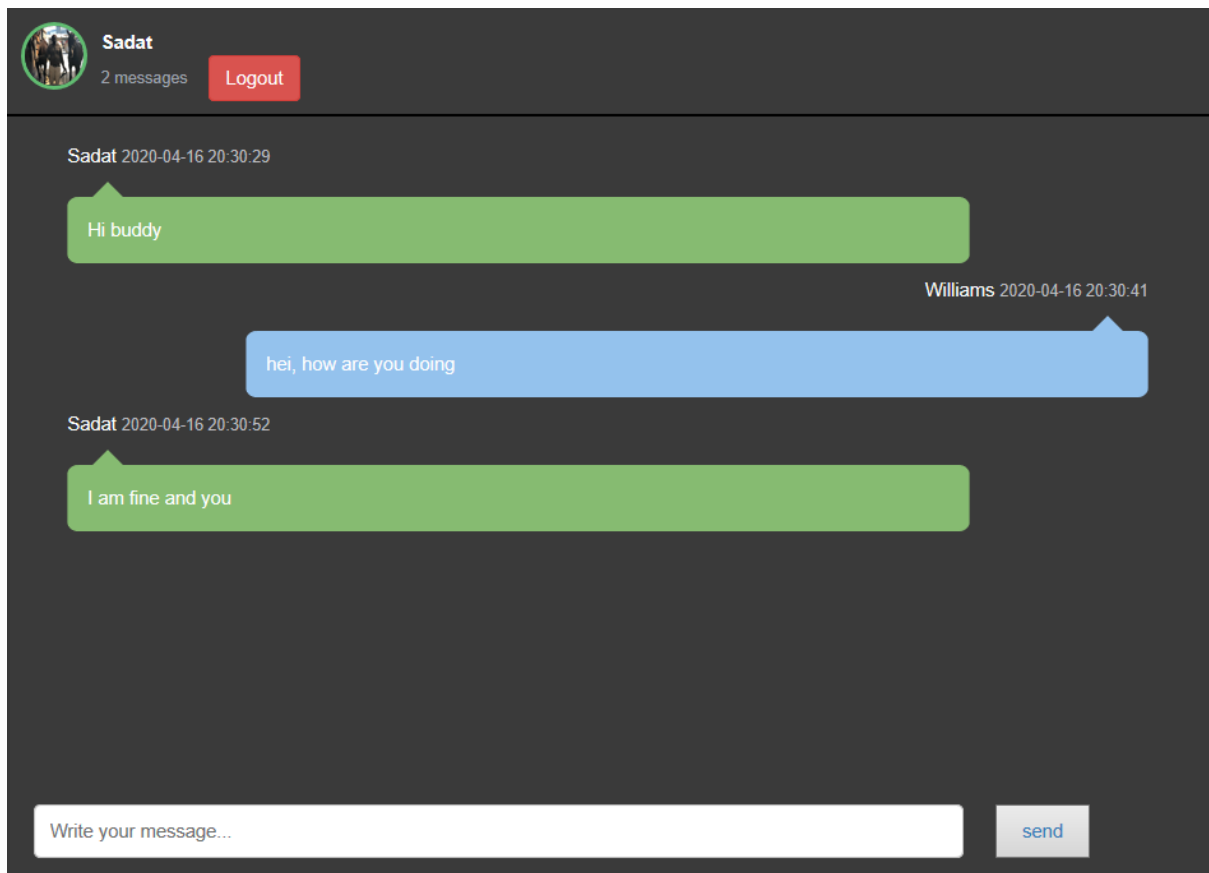
4.5.8 Chat Environment

This environment allows a user to select who to chat with. The chat in this environment happens synchronously and in real time. The images below shows a real time conversation between two members, Sadat and Williams.
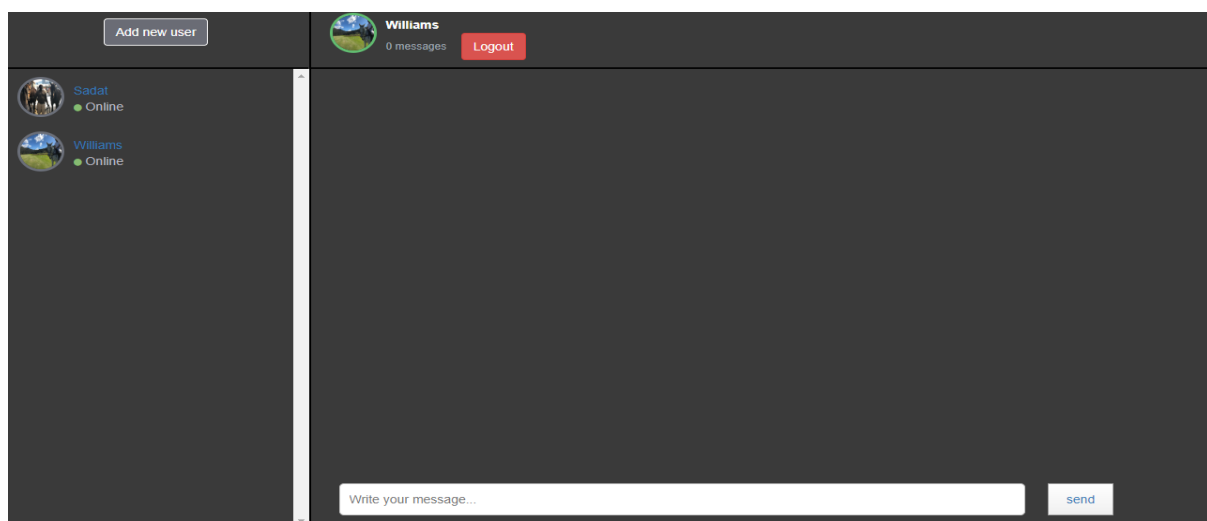
Figure 10.1.9 showing chat area

Figure 10.2.0 showing a chat with a member

Figure 10.2.1 showing a chat with a member.

4.6   **Application Chatbot**

Since the application consists of many functionalities, it becomes prudent to have a chatbot that will help users navigate the platform. Using Landbot.io; a SaaS platform that facilitates automated interaction between a firm and its clients, a chatbot with navigation features is implemented to aid user navigation on the application.

The figure below depicts the application homepage a user visits for the first time. At the right side is the interactive bot to help guide a user.
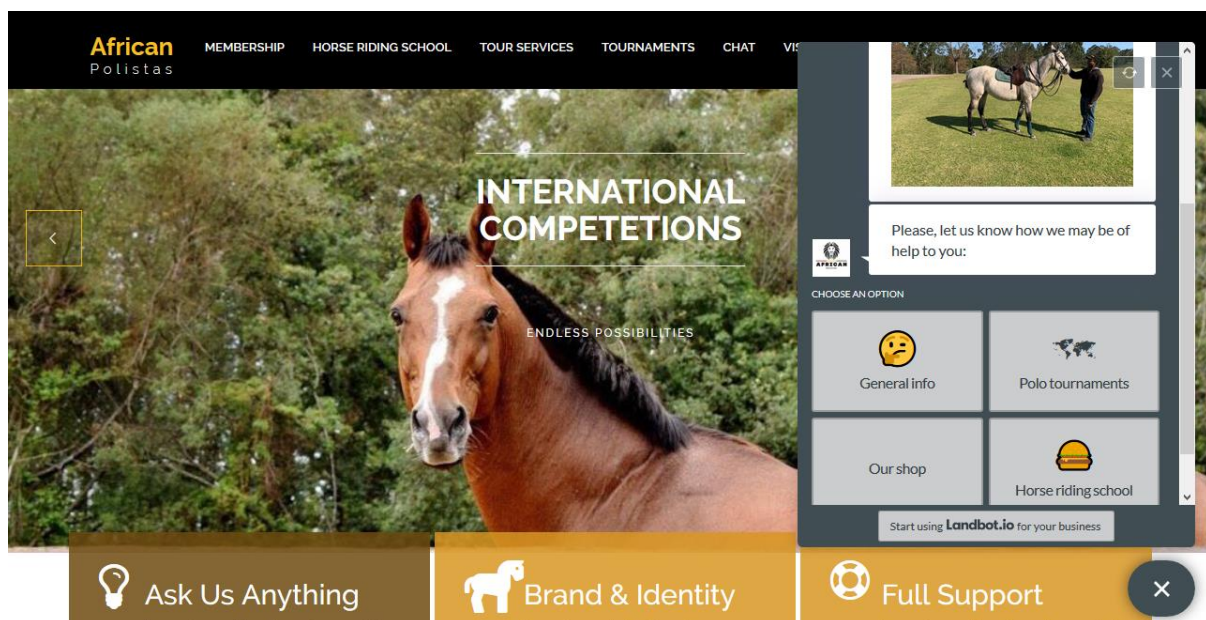
Figure 10.2.2 depicting welcome page with chatbot at right side

The image below shows details when a user clicks to view general information about the club.

This and many more interactions can be done on the system.
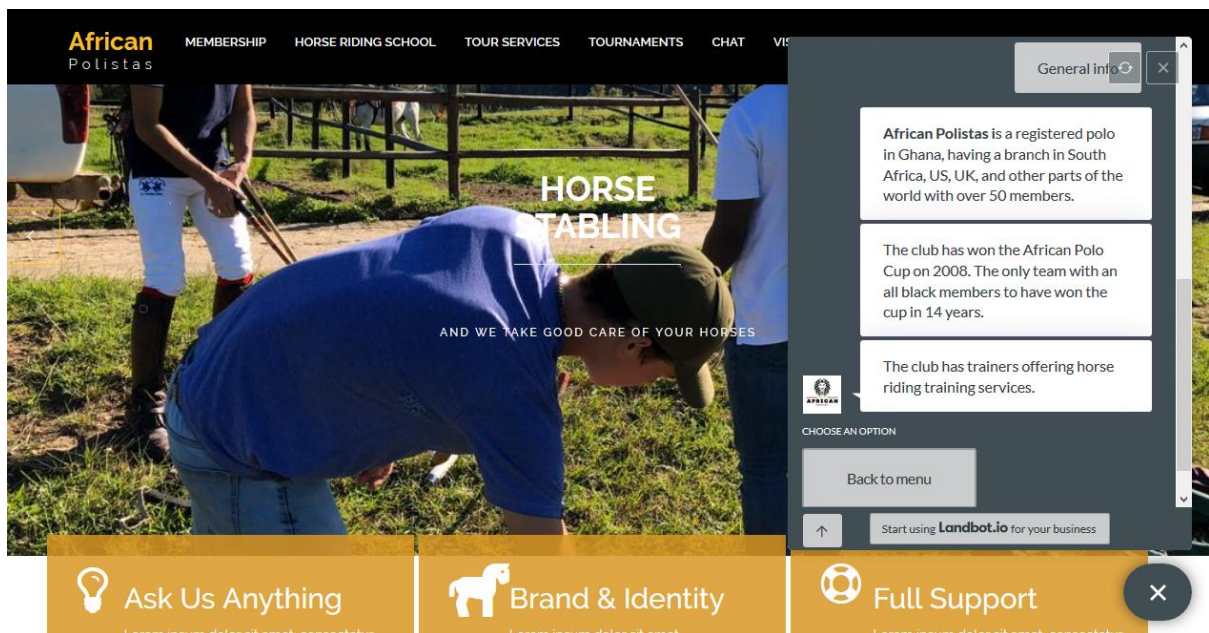
Figure 10.2.3: a user requiring general information about the club

4.7   System and User Testing

At this stage of the project, most functionalities of the application have been built. The major components of the project; the e-commerce store and chat platform are fully built and tested. Because of covid19 pandemic, I have not been able to get in touch with the club owners for them to test the application. What I have done is to get some learned members of my community to test the system. Before testing the system, I explained to them the idea behind the project. They all interacted with the system and expressed satisfaction with the systems usability. With the e-commerce platform, they interacted with it until the purchasing point where a successful purchase was made. Some of them made mobile money purchases whilst others used dummy card details provided by flutterwave for the payment. It was all successful. With the chat platform, all the users were made to make membership registration requests, once accepted, password reset email is sent to their mails. They all logged into their emails on my laptop to access the links, they created their accounts and were able to chat with each other. They all expressed satisfaction with the system and hope that the interface can be improved. I engaged with the club owners through Zoom video conferencing to enable them to have a feel of the application. They seem excited about the development and have some concerns about the interface of the application. Overall, the systems works fine as envisaged from the beginning of the project.

Chapter 5: Conclusion and Recommendation

5.1 Future Works

Parts of the future work of this project is to improve the security features of the chat platform. As it is currently, the chat between members is not encrypted, any hacker who is successful in getting access to the database or become a man in the middle between two users can see the chats in plaintext. Future works will require the implementation of an end to end encryption algorithm to secure the chats between members.

Another future work will be to integrate a weather forecasting API to always display weather details of the training environment. This will inform the decision of members who want to go for training.

Currently, the application is web based, a future work will be to implement a mobile version of the application. Considering the programming architecture used, it should be relatively easier to implement a mobile version.

5.2 Limitation

The application can only be accessed via a web portal. Persons without access to the web cannot be able to use the system. With regards to the e-commerce platform, the club owners do not have a clear policy regarding delivery of products, so the e-commerce portal does not have a delivery management feature. Since I will continue work on the project beyond capstone, I will implement those as required and other functionalities that may arise.

5.3 Recommendation

There is work to be done to improve the aesthetics of the application. This has been an observation made by the testers of the system. Management of the club must design a purchase

delivery policy to enable the developer include delivery functionalities onto the e-commerce site. An end to end encryption systems needs to be implemented onto the chat platform to improve on the application's security.

5.4   Conclusion

The purpose of this project was to build an integrated system to help manage the activities of African Polistas. The system is to have an e-commerce store, a chat platform secured for members alone, and other services members of the club can subscribe to. E-commerce store is fully built, integrated with online payment system. An interactive chatbot API integrated to aid user interaction on the application. Polistas chat platform is fully built but work needs to be done to ensure end to end encryption of the chats between users. Members who wish to subscribe to services such as horse-riding training can do so on the platform. About five people were made to test the application and expressed various levels of satisfaction with how the application works. Generally, the major functional requirements of the application have been met. Through my discussions with the client, the client seems excited about the progress of the application and looks forward to having the application fully functional.

References

[1]   Stephen M. Marson PhD, ACSW (1997) A Selective History of Internet Technology and Social Work, Computers in Human Services

[2]   John L. Scott & Ruth Miller (2002) Electronic Commerce Encourages Differentiation, Journal of Internet Commerce, 1:4, 1-21

[3]   Richard C. Latham. Polo Sport Encyclopaedia Britannica. 2019

[4]   Patricia Lanford and Roland Hübscher. 2004. Trustworthiness in e-commerce. In *Proceedings of the 42nd annual Southeast regional conference* (*ACM-SE 42*). Association for Computing Machinery, New York, NY, USA, 315–319. DOI:https://doi.org/10.1145/986537.986614

[5]   Jim Conallen. 1999. Modelling Web Application Architectures with UML. *ACM,* 42, 10. https://dl.acm.org/doi/pdf/10.1145/317665.317677

[6]   Dragus-Paul and Adam Alter. 2014. Designing an MVC Model for Rapid Web Application Development. https://doi.org/10.1016/j.proeng.2014.03.106

[7]   Asbjorn Folstad and Petter Bae Brandtzaeg. Chatbots and the New of HCI. Volume 24, Number 4 (2017), Pages 38-42. 10.1145/3085558

[8]   Tim Wright. 2009. HTML5 and the Future of the Web.   Retrieved from: https://www.smashingmagazine.com/2009/07/html5-and-the-future-of-the-web/

[9]   Rachel   Andrew.   2019.   How   to   Learn   CSS.   Retrieved   from: https://www.smashingmagazine.com/2019/01/how-to-learn-css/

[10]   ZURB. 2010. Spicing Up Your Website With jQuery Goodness. Retrieved from: https://www.smashingmagazine.com/2010/06/spice-up-your-website-with-jquery-goodness/

[11]   What if PHP? Write Your First PHP Program. https://www.guru99.com/what-is-php-first-php-program.html

[12]   Margaret Rouse. 2018. MySQL. https://searchoracle.techtarget.com/definition/MySQL

[13]   https://www.cs.wcupa.edu/rkline/index/xampp.html

Appendices

Appendix 1

Figure 7.0	showing use case diagram of system administrator
Figure 7.1	showing use case diagram for users
Figure 8.0	depicting client server architecture interaction.
Figure 8.1	depicting the interaction of MVC Architecture.
Figure 8.2	showing the landing page when the application is visited.
Figure 8.3	depicting the home page of the e-commerce platform
Figure 8.4	depicting home page of the chat platform
Figure 8.5	depicting administrator dashboard
Figure 8.6	showing entity relationship between the various tables in the database.
Figure 9.0	showing code snippet of database class.
Figure 9.1	depicting session class.
Figure 9.2	showing admin login page
Figure 9.3	showing login fail for admin with wrong credentials
Figure 9.4	depicting the administrator dashboard.
Figure 9.5	depicting add product brand page in the dashboard