**Online Shoe Store Application**

**Introduction**

Shoes are a vital aspect of human life that are primarily used for protection against injuries and different forms of infections. The project will involve establishing an online shoe store that will meet the needs of the people locally as well as internationally. Technology has revolutionized different aspects of human life. Additionally, almost everyone nowadays has access to internet, therefore developing an online platform would meet the specific needs of the consumers.

**Purpose of the study**

The online shoe store will involve a collection of shoes according to different gender, age group, and professional. This essay will focus on describing the overall overview of the online shoe store.

The main goal of establishing an online shoe store is to ensure quality, convenient and affordable access of shoes to every individual locally and internationally. Additionally, the store will ensure consistent high-quality product that will meet the standards of the population.

**Significance of the study**

This shoe shopping Application is a good collection of footwear for girls and boys paired with extremely profitable offers and deals to choose from this application. Thus, this online shoe shopping application will relieve people by providing access for purchasing shoes just by sitting at home.

This online shopping application has two modules namely, Admin and Customer. Admin can add shoes, view products, view customers and also admin can view the customers’ orders. Customers can register and login using credentials. He/she can view products, can add products to the cart and do the payment, they can also track their order and view their previous order history.

The Shoe store application will be built using a number of technologies namely;

|  |  |
| --- | --- |
| **Front End** | **Back End** |
| HTML, CSS | Python3 |
| React JavaScript | Django |
| Redux (State Management) | MySQL database |

**IDE:** Visual Studio Code

**Architecture of the Application**

The Application proposed will follow the MVT (Model View Templating) pattern which is what Django uses for its project structure.

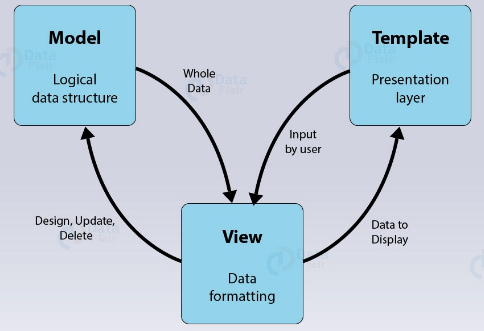
Django is based on the MVT (Mode-View-Template) architecture for developing a web application.[[1]](#footnote-1)

MVT Structure has mainly three parts

**MODEL:** This is where an OOP pattern is followed to create a class which will act as our model or template to communicate with the database directly. The model acts as an interface for our data and is responsible for maintaining data. It is the logical data structure behind the entire application and represents the database. In this current project, we shall make use of SQL queries to make request to and from the database.

**VIEW:** This represents the user interface; it connects the model to the template and here is where our API logic that communicates from model to template will be written.

**TEMPLATE:** This consists of the static parts of the desired HTML output as well as a special syntax describing how dynamic content will be inserted. For the case of Django, we shall use the Django Templating engine as our syntax to write our logic for the templates.



A user requests for a resource to Django and Django works as a controller and checks to the available resource in URL.

If the URL maps, the view is called that interacts with the model and template so that the template is rendered.

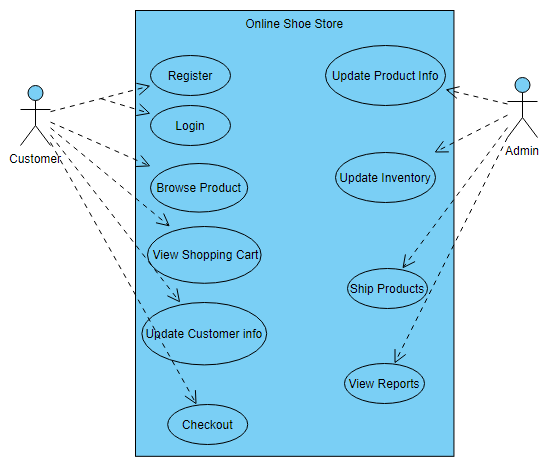
Django then responds back to the user and sends a template as a response.

**Modules:**

The system comprises of 2 major modules with their sub-modules as follows:

1. **Admin:**
   * **Login:** Admin can login in his personal account using id and password.
   * **Add Shoes:** Admin can add a shoe.
   * **View users:** Admin can view the users.
   * **View User order:** Admin can view the order place by user.
2. **User:**
   * **Login:** User can login his account using id and password.
   * **View products:** User can view the products.
   * **Add product:** User can add desire product in the cart.
   * **Track order:** User can track is order.
   * **View Order history:** User can see his/her previous orders history.

**System Model – Use case diagram**



**Use case description**

**Register:** If the customer is a new user, he can request to register page. A register page opens up asking for information about the customer and after successful registration, customer is automatically logged into the application.

**Login:** The customer can login by entering his username and password and the system verifies if password matches otherwise, it shows error message to the customer.

**Browse products:** The customer requests to view the products in product category and the system will display information about products in the selected category.

**View Shopping cart:** The customer requests to view the shopping cart and the system returns the shopping cart to the customer to view the product pricing and total prices of items in the cart.

**Update Customer info:** The customer requests to update their information such as name password, and delivery address, if updated successfully the information is also updated in the database of the system.

**Check out:** The customer completes the shopping request to checkout. If the payment information exists, then the credit card information is sent to the credit card company. Once there are enough funds on the card, and card is valid, the order is processed by the system and the checkout process is completed.

**Ship products:** After getting the order request from the customer by the admin, the admin ships the ordered product to the customer’s address within 5 to 6 working days.

**Use case Text**

* Customer is presented with the index page where he can view a variety of shoes displayed under their different categories.
* Customer then selects one of the shoe products he or she desires to purchase and adds the desired product to his cart.
* The customer is then requested to login where by if he is a new user, he is able to register and provide all his information including but not limited to name, email, and delivery address.
* If the customer is an existing user, he/she is prompted to login with his/her credentials and if they have forgotten, there is a provision to reset one’s password, the systems will send a reset link to their email.
* Once the customer has successfully logged in, he/she then proceeds to the checkout page where he/she is requested to select a payment method such as mobile money payment or card payment.
* Once transaction is successful, a receipt is generated, confirmation email is sent and the product is delivered to the customer with in 6 working days.

1. <https://www.geeksforgeeks.org/django-project-mvt-structure/#:~:text=Django%20is%20based%20on%20MVT,for%20developing%20a%20web%20application.&text=View%3A%20The%20View%20is%20the,CSS%2FJavascript%20and%20Jinja%20files>. [↑](#footnote-ref-1)