**CHAPTER IV: SYSTEM IMPLEMENTATION**

**4.1. Implementation and Coding**

**4.1.1. Introduction**

The Online Shoes Storing Management System implementation stage refer to a process of translating the design specifications into a working system. This chapter gives details of coding practices, tools, and technologies that were used during the stage of implementation. It talks of everything from the initial set-up up to the eventual launching of the system, detailing the major development milestones and the work of integration.

**4.1.2. Tools and Technologies Used**

The Online Shoes Storing Management System has been developed with the following tools and technologies:

**Programming Languages:** This system was primarily developed using PHP scripting on the server side, JavaScript for client-side interactivity, and SQL for managing its database.

**Database:** MySQL is used because of its powerful data management, and PHP can be easily integrated into it.

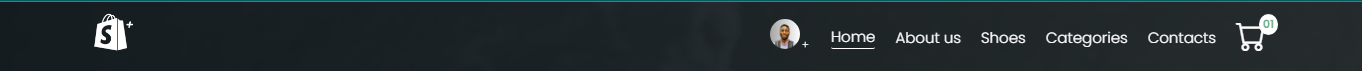
**Development Environment:** The system development was carried out in Visual Studio Code. Version control: Git.

**Hosting and Deployment:** The application has been hosted on a Cloud service provider that deployed the Apache server to ensure better scalability and reliability.

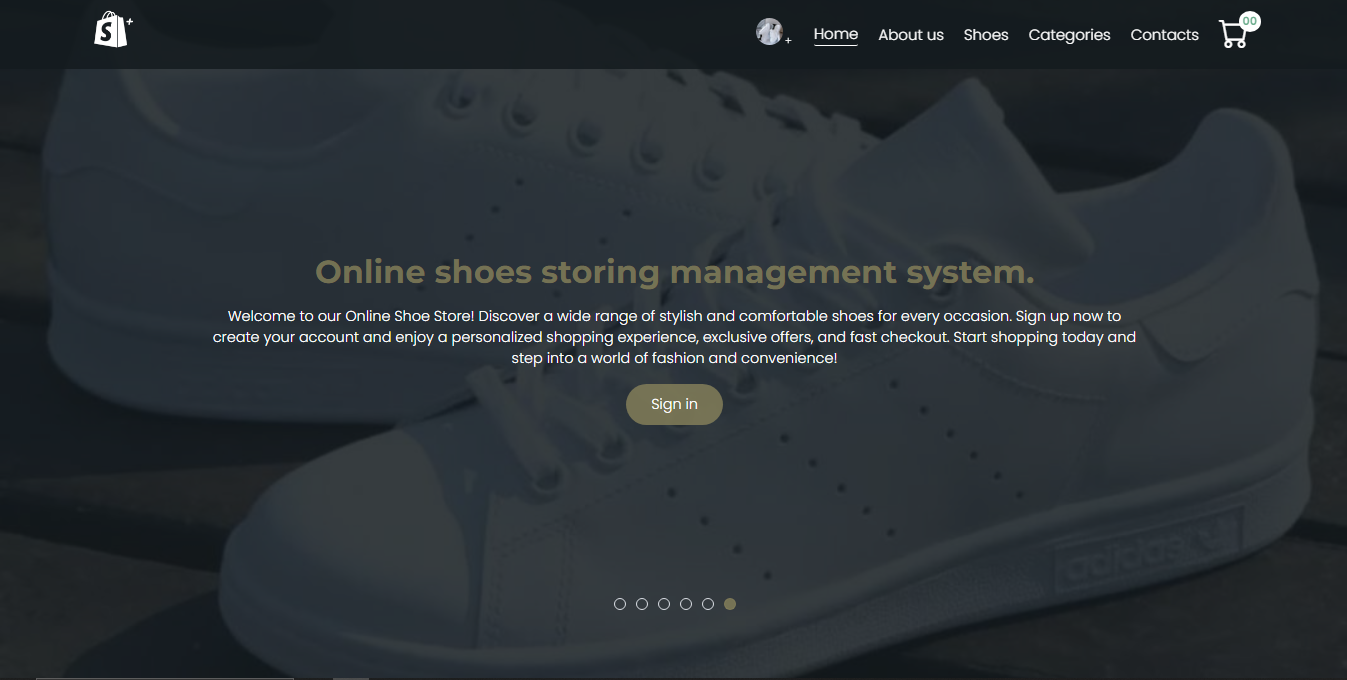
**4.1.3. Screenshots**

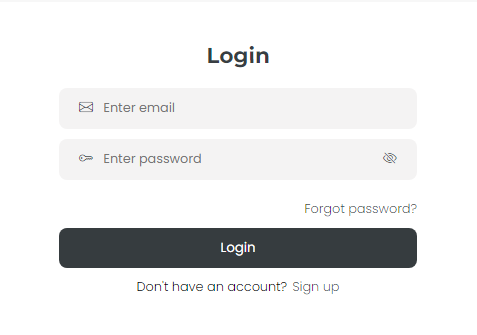
**4.1.3.1. User side**

**Menu**

****This menu bar lists the links that redirect to pages and it redirect to the cart in order to view shoes that are desired to be purchased by the user and it has the profile photo icon of the user when connected. That icon has a dropdown that redirect to the user profile, dashboard and payment history. If a user connected is an admin, it shows a link that redirect to the admin page

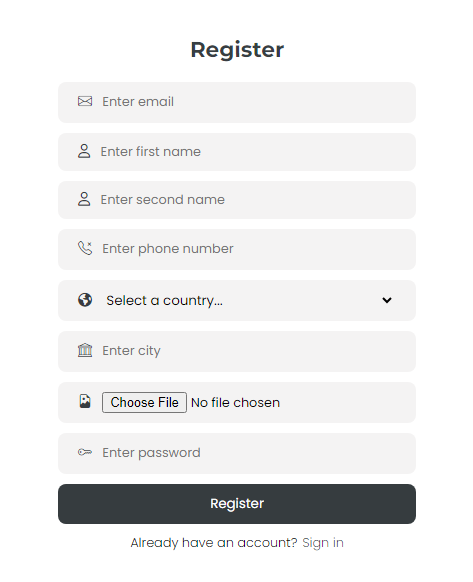
**System slides**

****This system slides first it welcome the user to the system, containing the system name and the sign in button. It contains shoes slides and those slides are added at the admin part, those slides are the new shoes collection daily.

**Login   
**

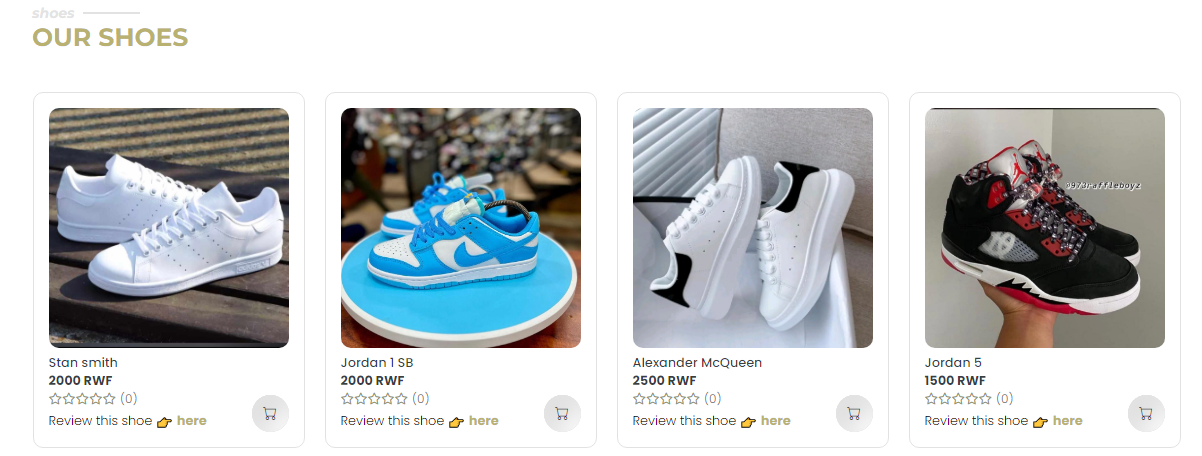
This is a login page where a user will enter his email and password to access to his account and manage it. If he accesses to this page and he has no account he can click on sign up to create an account and if he has forgotten his password he can click on forgot password where it will redirect to forget password to page and enter his email to receive a link for resetting the password.

**Register**

****

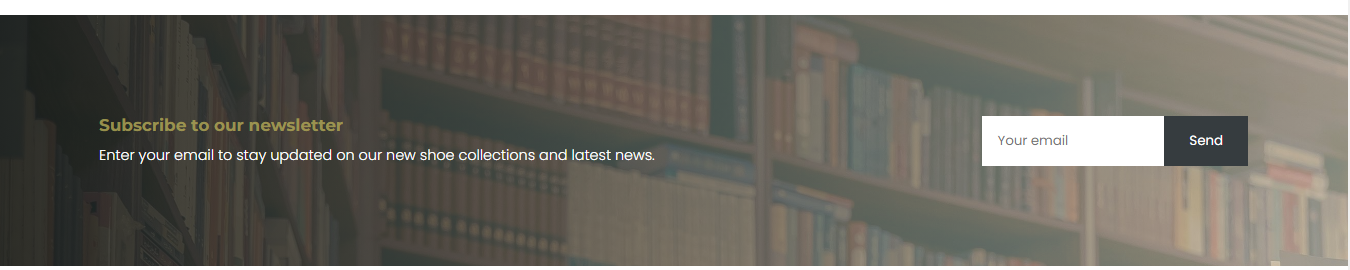
This register page is where the user will complete the fields to create his account. If he has an account already he can sign in or login .

**Shoes**

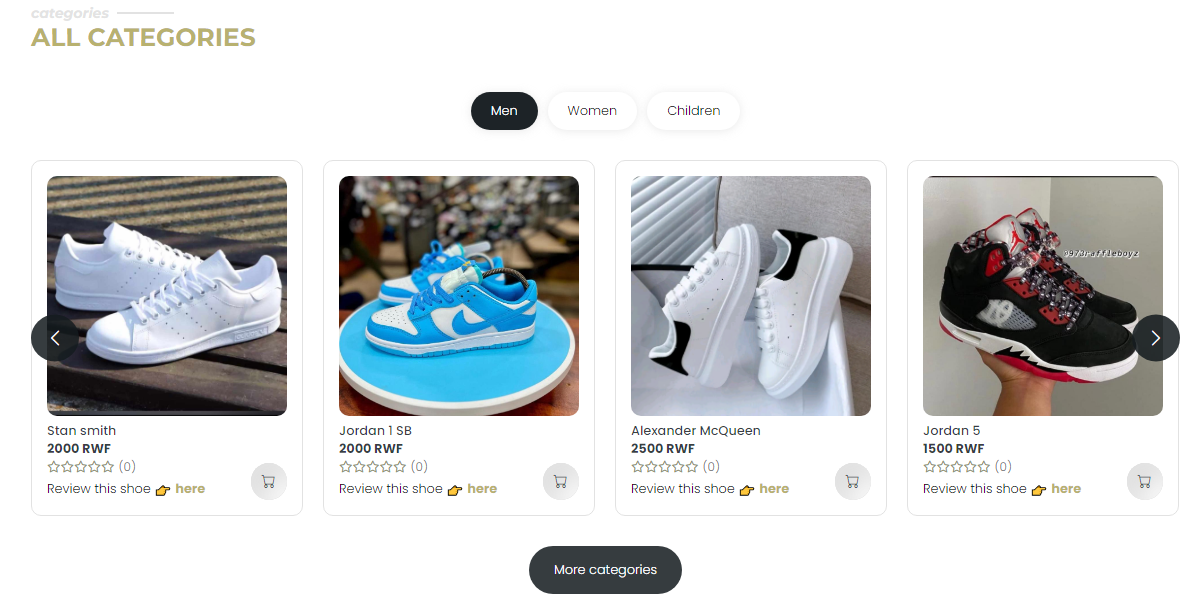


This part list 8 recent shoes added in the system each shoe with the name, price, the rates, the link that redirect to the rating page and the link that redirect to cart. Under the shoes there is a button that redirect to view in details when you want to view many shoes

**News letter**

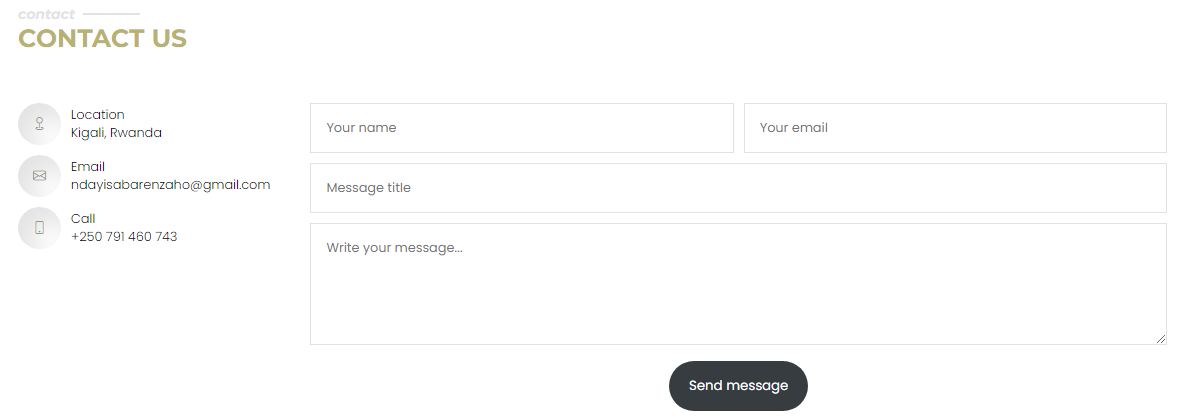
****The newsletter part is the part where a user will enter his email and his email will be stored in the database to receive updates on new shoe collection and last news.

**All categories**

****

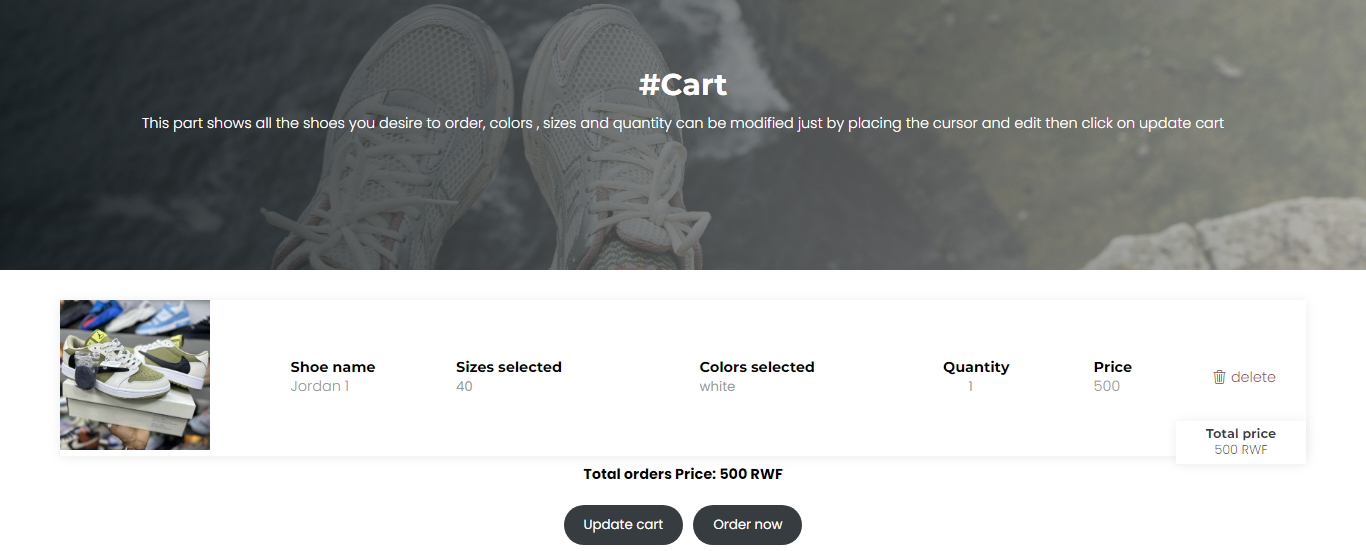
This part contains three main categories men,women and children . it filter shoes according to type, ech shoe containing the name, price, rates and the link that redirect to the rating page . if a user want to view more different categories , he will click on more categories

**Contact us**

****

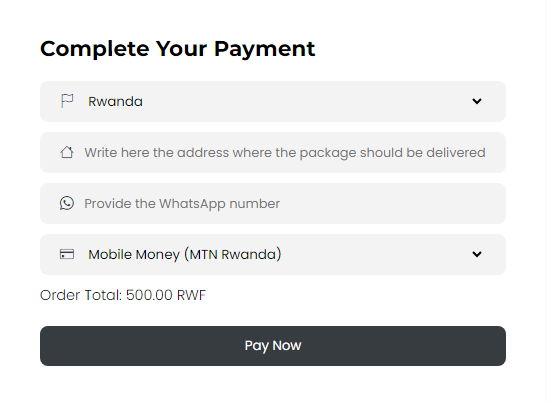
This contact part is where the user will enter his identity to contact the admin and the message will be received by the admin when sent. It can be either a message of demanding a shoe or a suggestion message and so on.

**Cart**

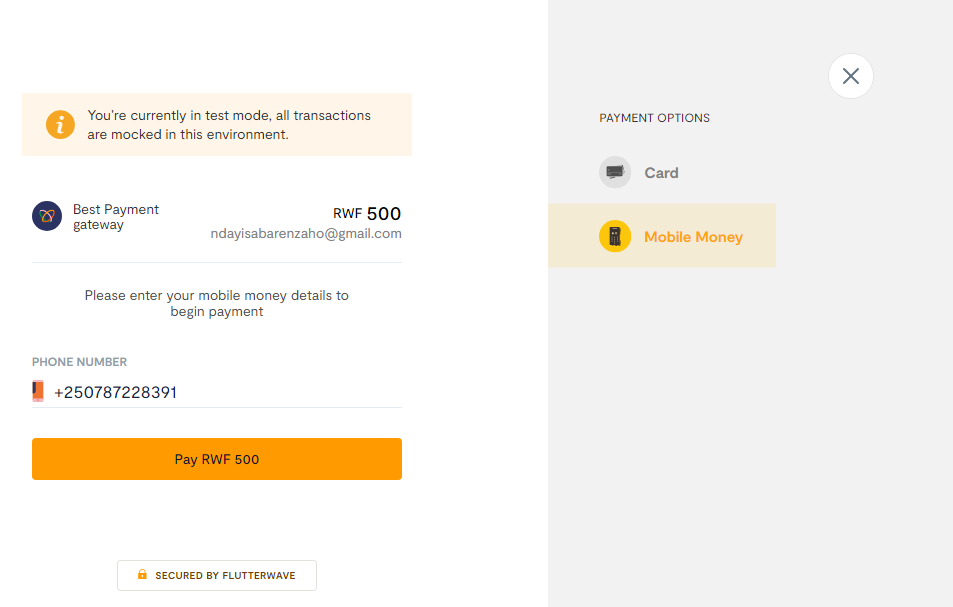
****

This cart contains the shoes that are desired by the user to be ordered, a shoe has details like sizes, colors and quantity that can be updated. A shoe in cart can be deleted when . The button update cart is for updating cart after increasing the quantity of the shoes desired to be purchased and the order now button is for placing the order and it redirect automatically to the payment page

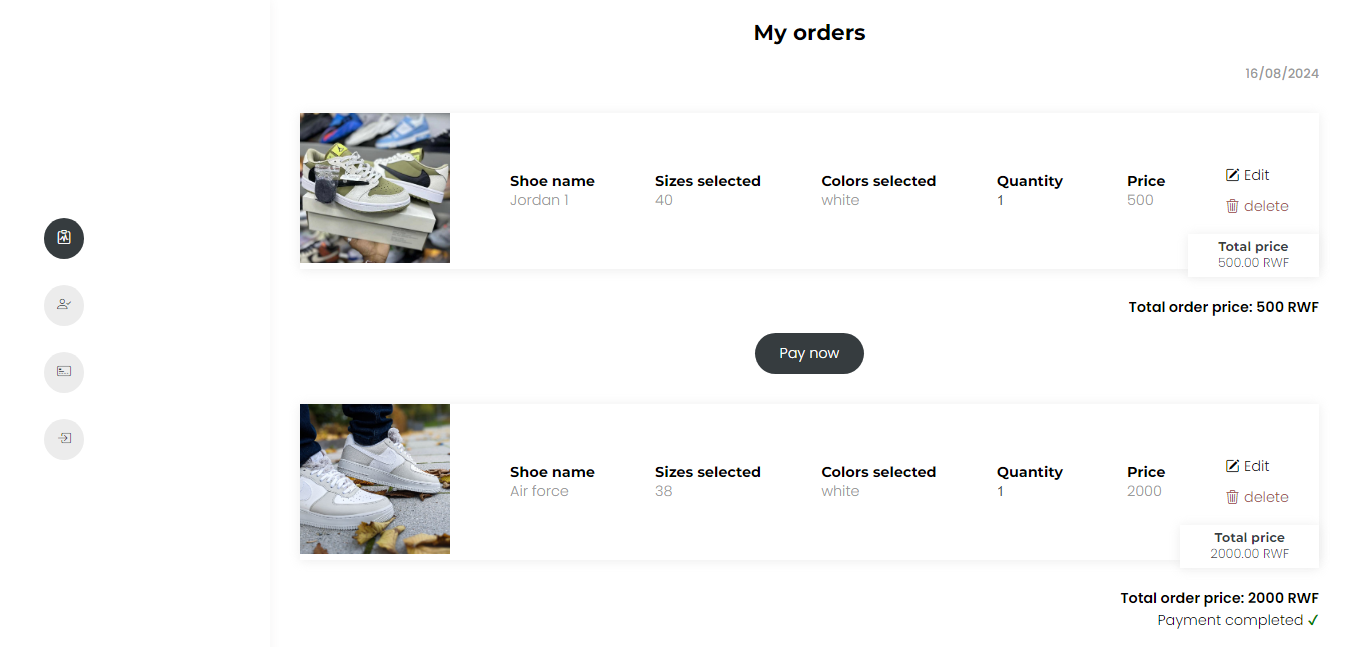
**Payment page**

****

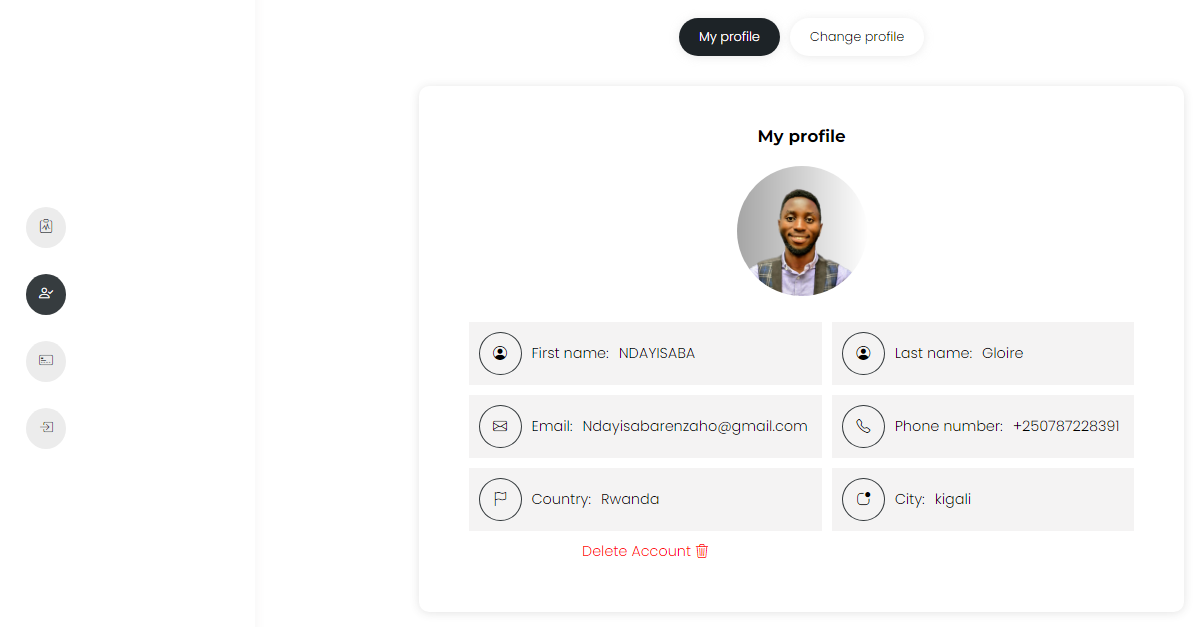
After placing an order a user can pay by selecting the country and the address to be delivered. The user can choose if his order will be shipped by choosing the country different from Rwanda and the quantity has to be 20 or more than twenty in order to be shipped out of Rwanda.

****

The payment that is implemented in this system is called fluter wave, it fetches the email of the user that will receive a successful payment and the mobile number that will be used to pay. The user can choose eithetr to pay with Bank using card or mobile money. **Users Orders**

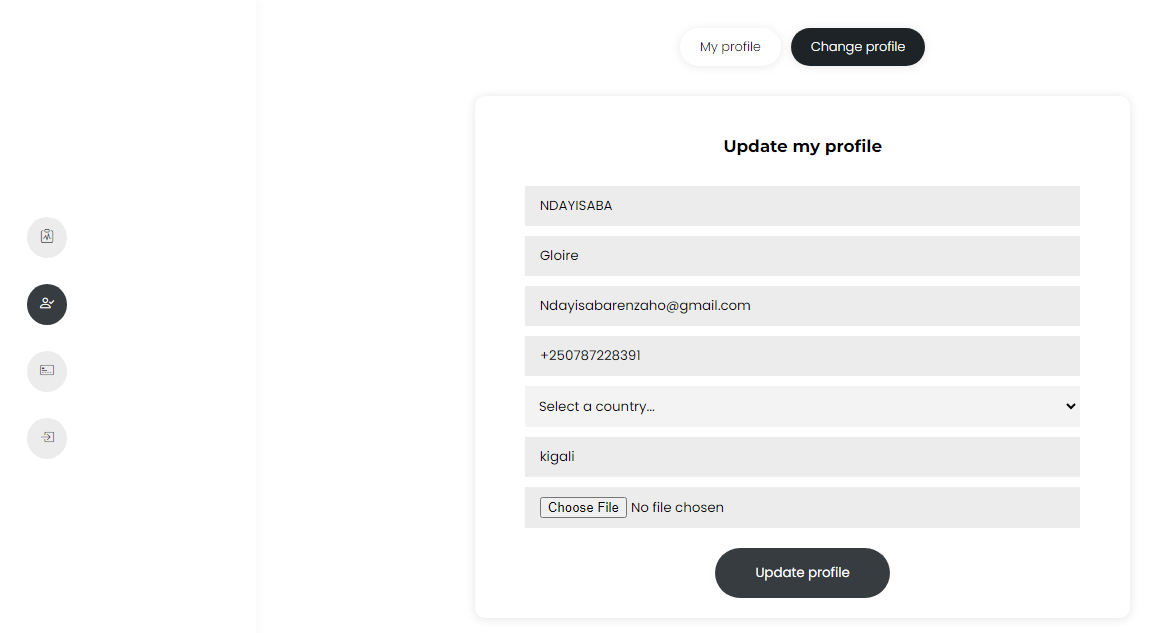
****

User dashboard contain all the orders that are made by the user and they are ordered by date those orders can be paid or not. They can be deleted or modified. If an order is not payed yet it can be paid again**.  
  
User profile**

****

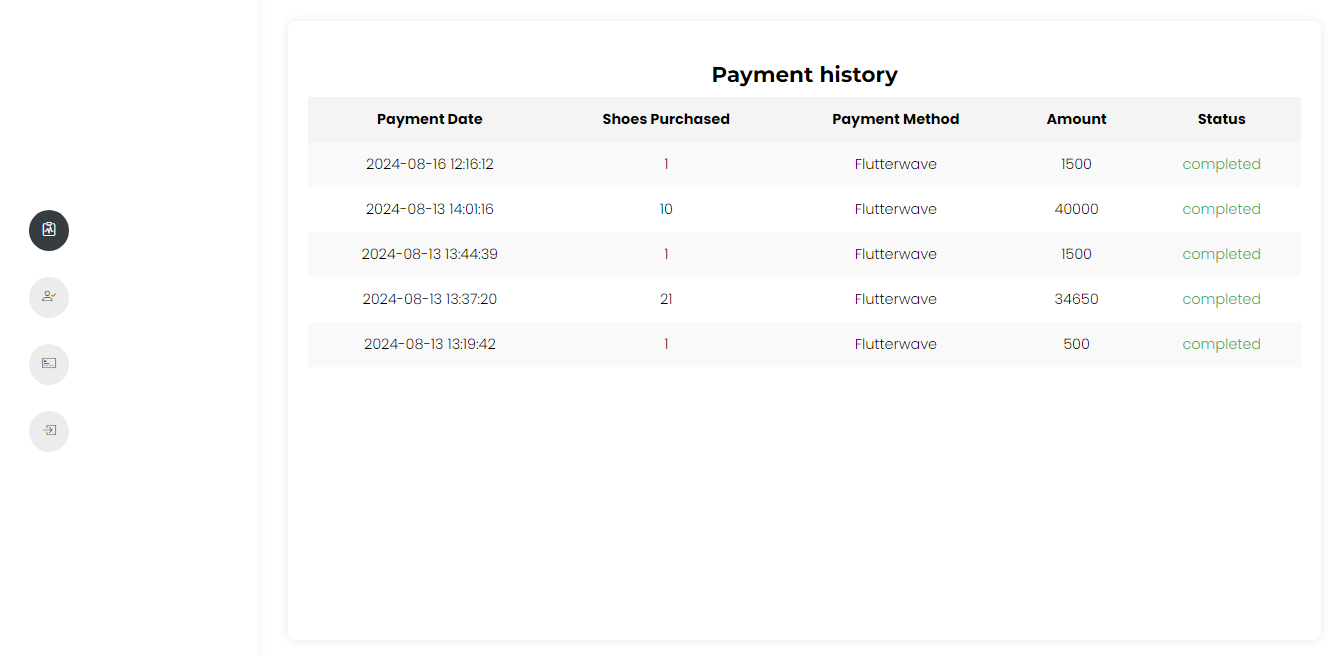
User profile shows the profile of an employee and his identity, a user can delete his account if he wants and to delete the account there is a popup that is opened to enter the password in order to confirm that the person who want to delete the account is the owner of it.

**Update profile**

****

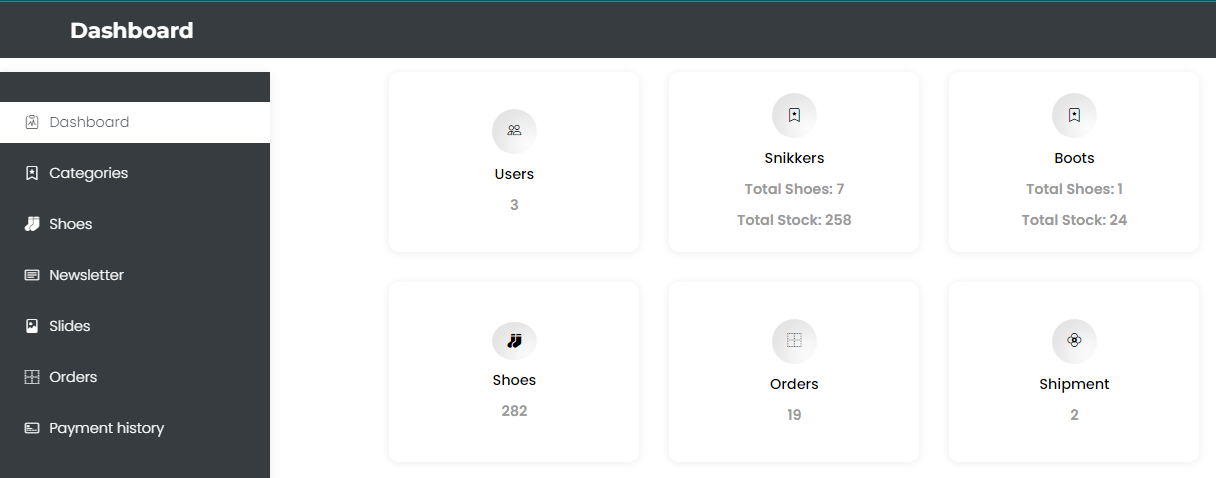
This part is where the user can update his account

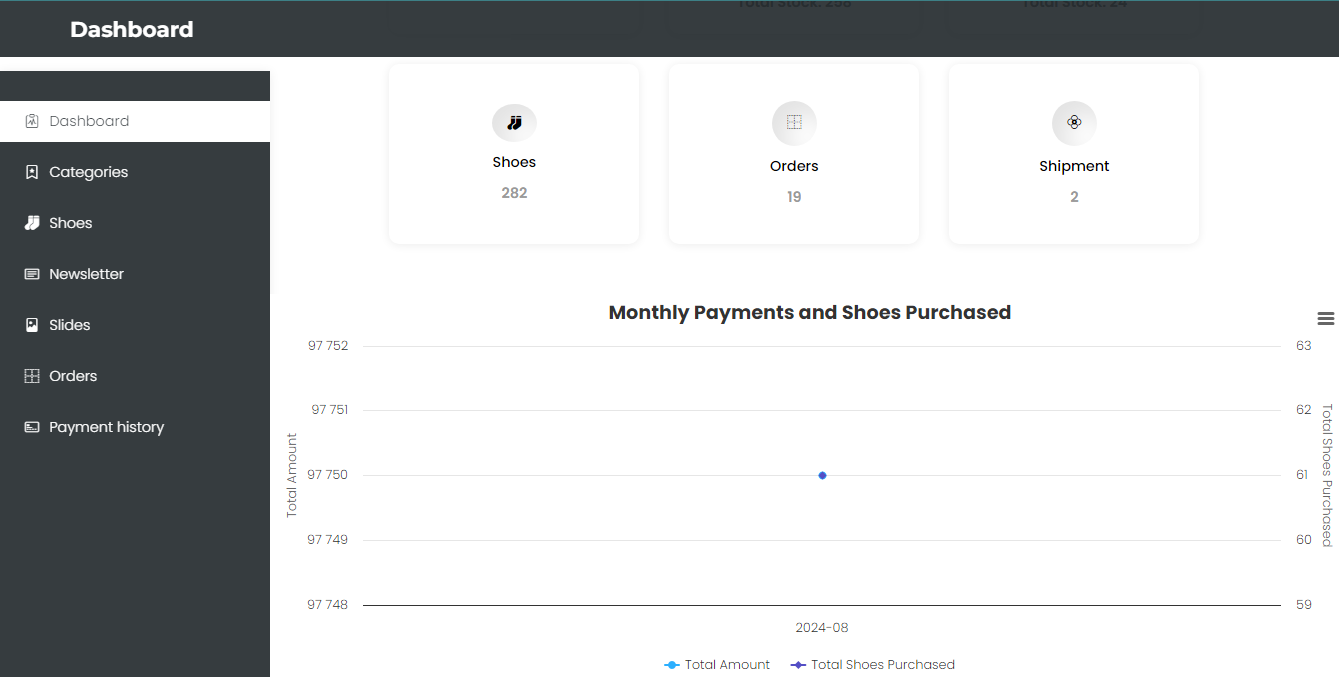
**User payment History**

****

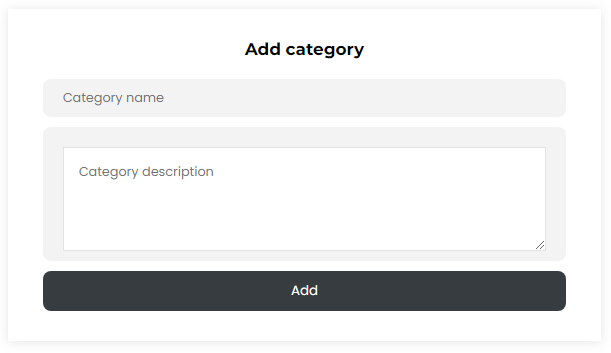
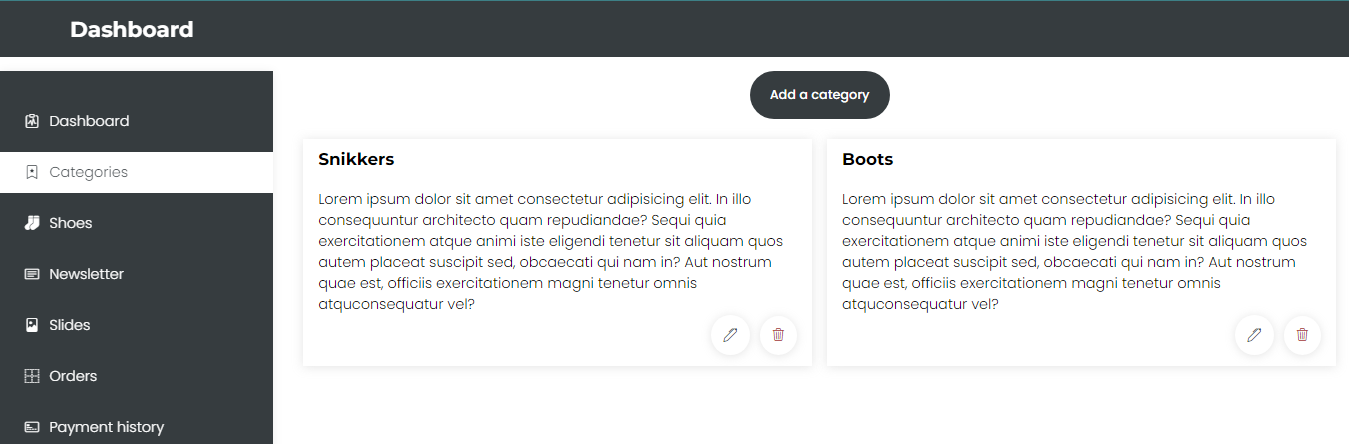
The payment history lists all the payment that has been completed, showing the date that payment was done and the number of shoes in the order, payment method, the amount and the payment status.

**4.3.1.2Admin Dashboard**

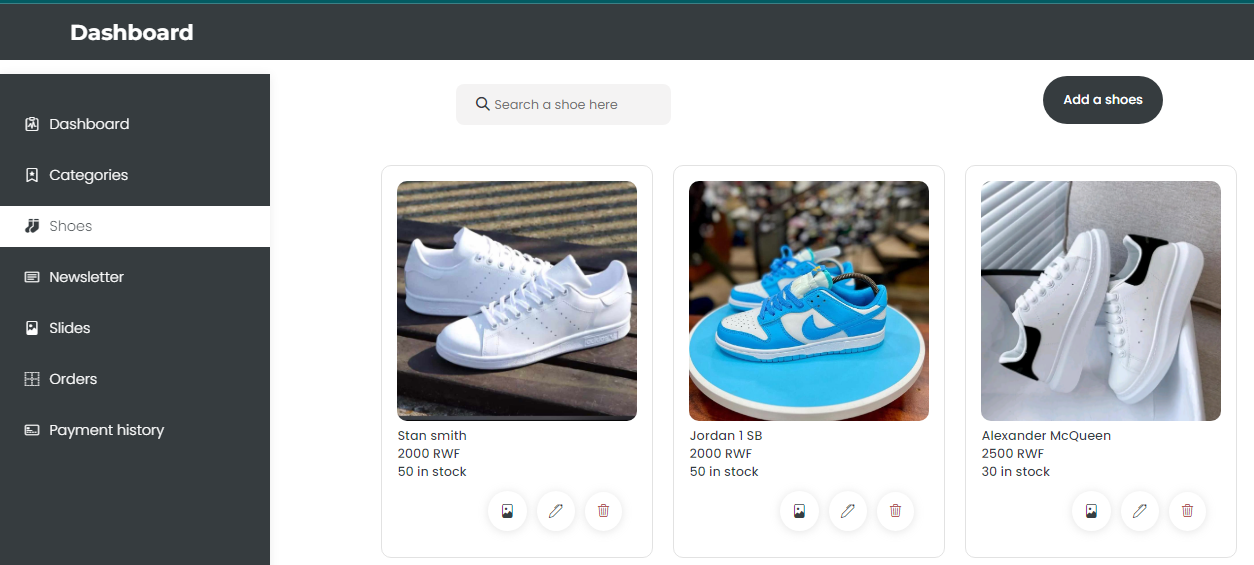
****

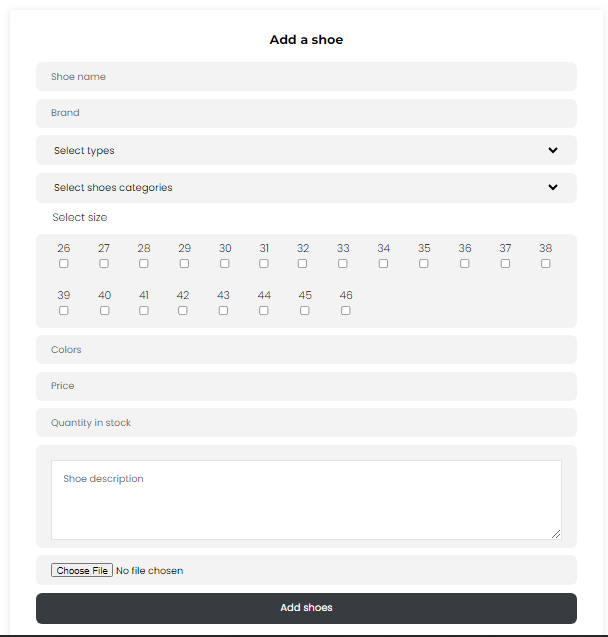
****

**Admin categories**

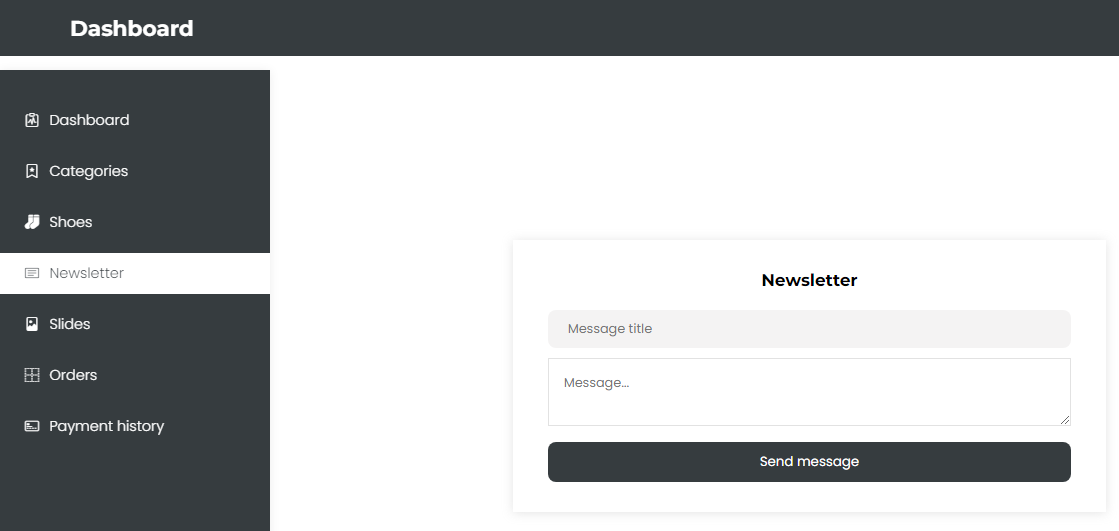
****

**Shoes**

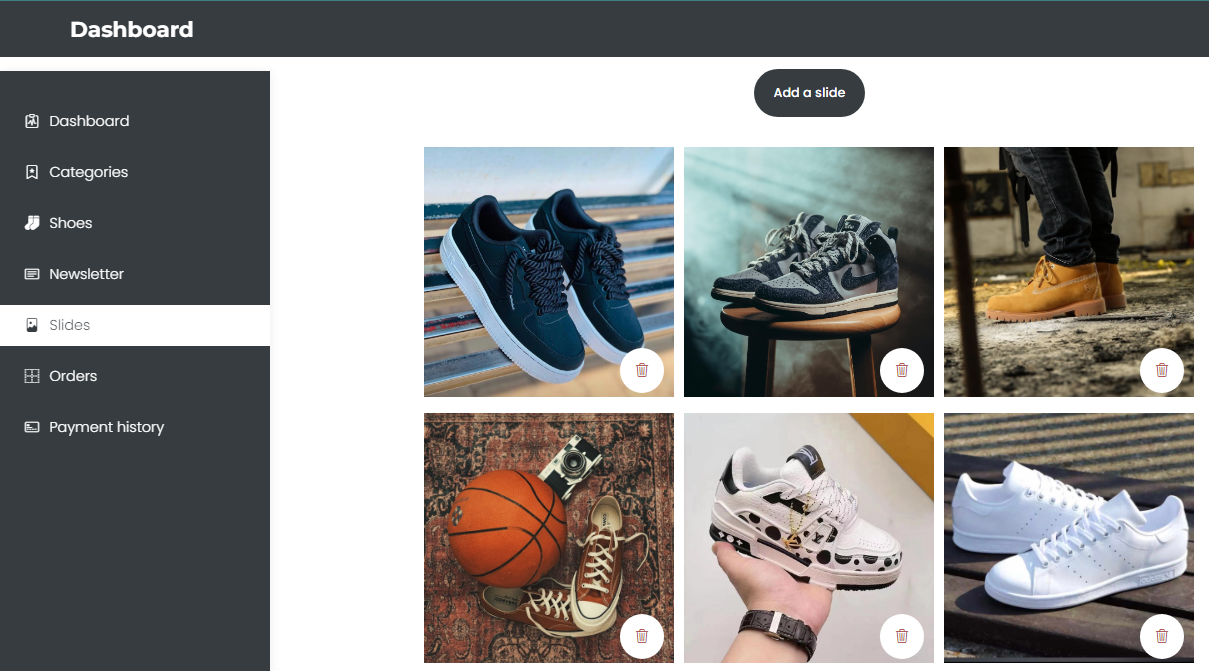
****

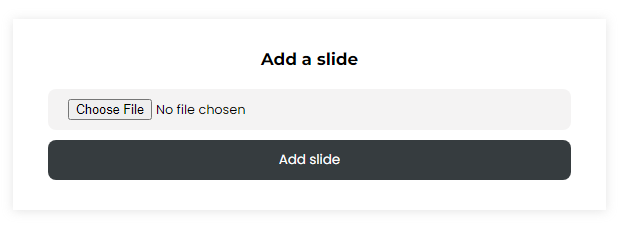
****

**Add a news letter**

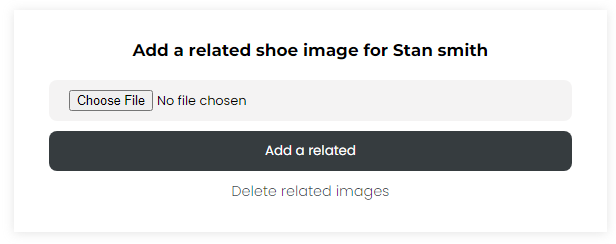
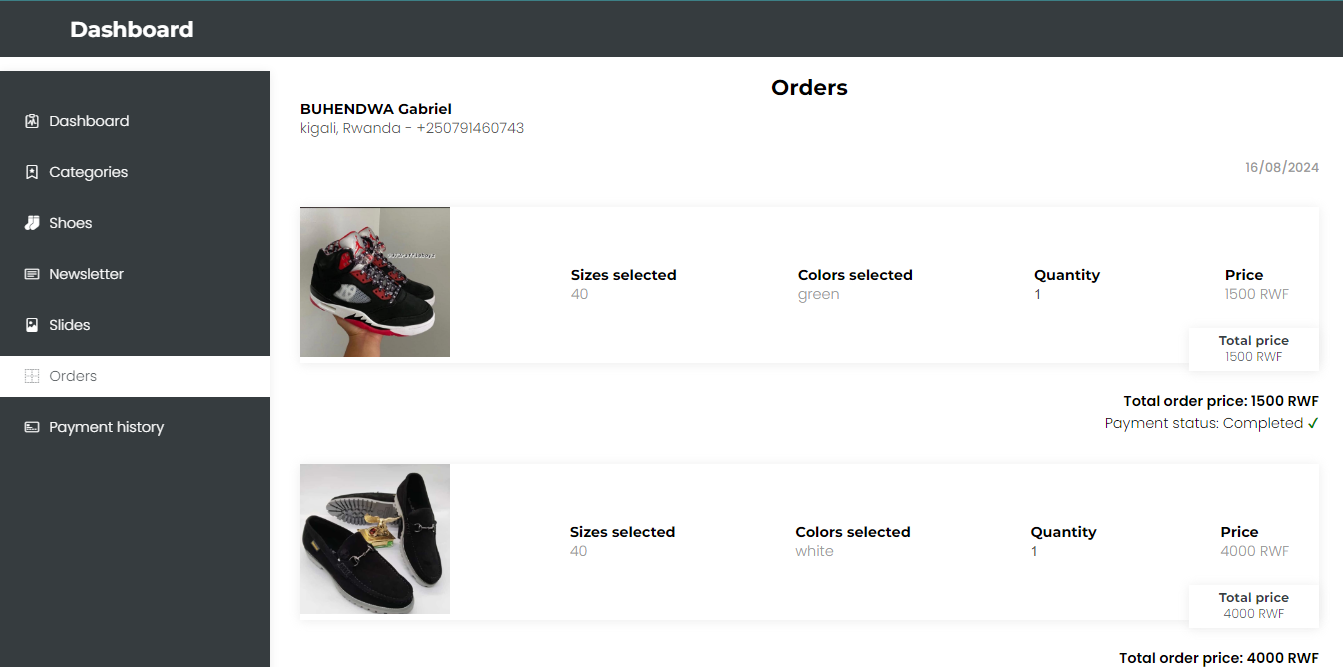
****

**Slides**

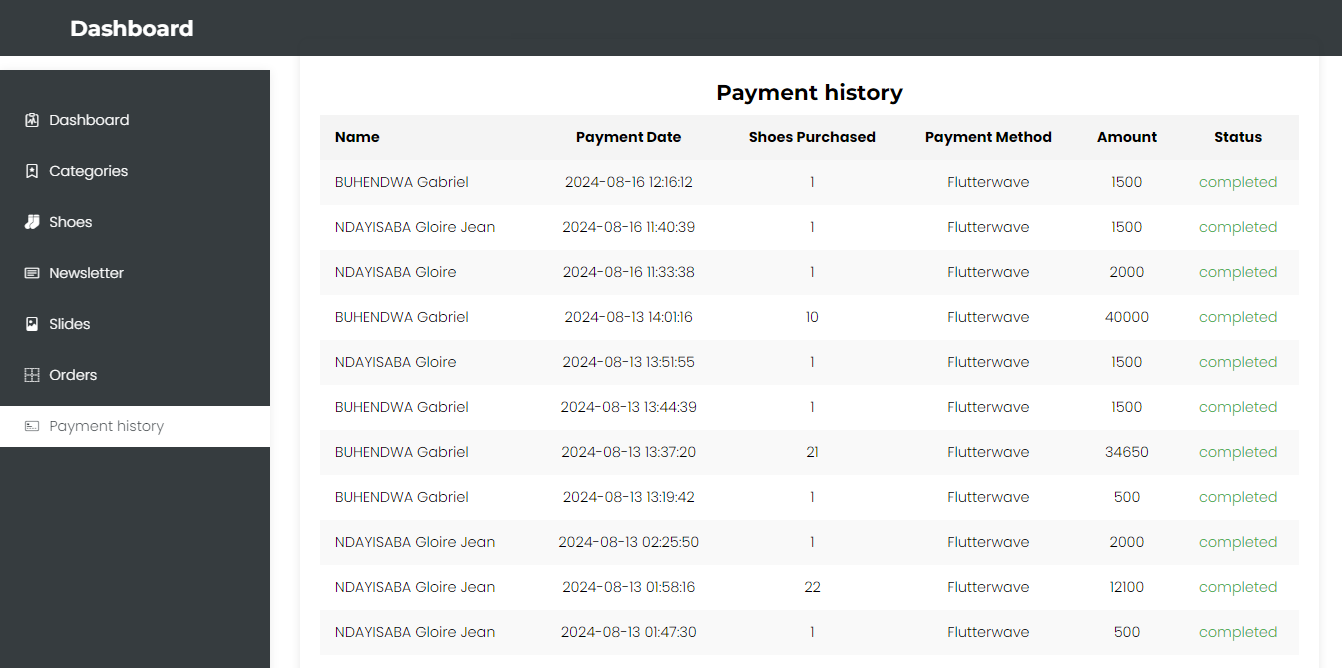
****

****

**Orders**

****

**Payment history**

****

**4.2. Testing**

**4.2.1. Introduction**

The Online Shoes Storing Management System will be tested to confirm that it was delivering the expected outcomes, as well as providing valid results. The testing will be carried out in steps in order to check whether the system is stable, as well as checking for performance and reliability.

**4.2.2. Unit Testing Outputs**

This section aimed at the different modules of the system as outlined below:

**Authentication Module:** Test results verified that user login and registration were successful.

**Order Processing:** Test results verified that the system was recording and processing orders successfully.

**Inventory Management:** Ensured the stock levels were updated correctly and accurately after the sales were made and after returns.

**4.2.3. Validation Testing Outputs**

The output of the validation testing showed that the system had met the user requirements:

User Interface: Its usability by the customers and it fulfilled the design requirements.

**Business Logic:** If all the business rules, in this case, price calculations and stock level updating, were implemented successfully.

**Data integrity:** This was to confirm that all data entries are properly validated and carry out without errors.

**4.2.4. Integration Testing Results**

Integration testing looked at the interface between different software modules that constituted the system:

**Frontend and Backend Integration:** A success in integrating the function between frontend and backend. The frontend and the backend are able to interact where the frontend sends requests to the backend, and the same is returned with the requested data.

**Payment Gateway:** Payment processing has been seamlessly integrated, ensuring proper recording of transactions.

**Email Notifications:** Sending auto-notifications are integrated with stock level changes and order confirmation.

**4.2.5. Functional and System Testing**

It was determined if each feature in the system worked according to how it was supposed to work:

**Search and Filter:** It is tested if customers can search for and filter shoes effectively.

**Shopping Cart and Checkout:** It is tested that shopping cart and checkout work accordingly.

**Admin Functions:** It is verified if admin features such as inventory management and monitoring sales work properly.

System testing tested the overall system performance:

**Load Testing:** Tested the system under several load conditions to verify that it was stable and responsive.

**Security Testing:** Conducted vulnerability assessments and responded to any possible security vulnerabilities.

**4.2.6. Acceptance Testing Report**

Acceptance testing enabled the final review of the system by the stakeholders:

**User Feedback:** User feedback was collected based on which it could be ascertained if the software had been developed as per their needs and expectations.

**Bug Reports:** Dealt with any issues that were not fixed during the acceptance stage.

**Deployment Readiness:** Checked whether the system was ready to go into production by assessing if it had been successfully tested and approved by stakeholders.