

**HCMC UNIVERSITY OF TECHNOLOGY AND
EDUCATION**

FACULTY OF INFORMATION TECHNOLOGY



Course name: IT Project

SNEAKER SHOP WEBSITE

Lecturer name: Mai Anh Thơ

List of members

Student ID	Student name	Contribution (%)
20110352	Nguyễn Việt Anh	100%
20110403	Lê Y Thiện	100%

Ho Chi Minh City, December 2022

Contents

PREFACE.....	4
I. Specific description	5
1. Software purpose:	5
2. Input data:	5
3. Use case:	5
4. Expected interface:	6
II. Work distribution:	6
III. Design	7
1. Algorithms:	7
2. Classes:	8
A. All classes	8
B. All Methods:	11
3. Database Desgin	18
4. Fields of each table:	19
5. Interface design.....	21
IV. Deploying and Testing:	28
Scenario 1: Signup/Log in before conduct buying.	28
Scenario 2: Conduct shopping.....	29
Scenario 3: Conduct shopping without signed in	30
Scenario 4: Admin sign in to	31
V. Conclusion	32
Pros:	32
Cons:	32
Future development:	32
Link source code:.....	33
Reference Source:.....	33

List of Figure

Figure 1 MVC model	7
Figure 2 Database	18

List of Table

Table 1 Use case	5
Table 2 Interface	6
Table 3 Work Distribution	6
Table 4 Author's class: Thien & Anh	8
Table 5 Author's class: Nguyen Viet Anh	9
Table 6 Author's Class: Le Y Thien	10
Table 7 Author's Method: Nguyen Viet Anh	11
Table 8 Author's Method: Le Y Thien	16
Table 9 List of table of database	18
Table 10 Field of User	19
Table 11 Field of Product	19
Table 12 Field of Invoice	20
Table 13 Field of Detail_In	20
Table 14 Interface Design	21
Table 15 Scenario 1	28
Table 16 Scenario 2	29
Table 17 Scenario 3	30
Table 18 Scenario 4	31

PREFACE

The first words allow the author to express sincere and deep gratitude to Mai Anh Thơ - the scientific instructor for the author, directed the topic, directly guided and had valuable suggestions to help the author during the research implementation. The author would like to thank the teachers and teachers of Ho Chi Minh City University of Technology and Education for teaching and imparting valuable knowledge to the author as the foundation for this study.

At the same time, the author would like to thank the Masters and Postgraduate Department of Ho Chi Minh City University of Technical Education for their enthusiastic help and guidance in completing the procedures related to the study. assist. The author also would like to sincerely thank you, sisters and colleagues for creating favorable conditions, providing documents, discussions to help the author complete the scales, surveys and take time to refer. and answer the survey.

Finally, the author would like to thank family and friends for their help and for creating a solid support for the author during the study and to obtain the research results.

Sincerely thanks.

Ho Chi Minh City, December 2022

Authors

Nguyễn Việt Anh

Lê Y Thiện

I. Specific description

1. Software purpose:

A responsive shopping website that allows customers to buy sneakers, shoes. Apart from that, administrator can manage the list of products on the website.

2. Input data:

We use Clear Database MySQL to stores list of products, Invoice and Users.

Request from website.

3. Use case:

Table 1 Use case

Order	Use case	Description
1	Login/ Logout	Sign in/Sign out to use the website.
2	Register	Create a new user and store information to database. Send an authentication code to user email to verify account.
3	Manage Product	Add, Delete, Edit a specific product by administrator.
4	Manage Invoice	Control the list of products which user has bought. Store customer invoices to the database
5	Order confirmation	Send a confirmation email contain bill to customer.
6	Cart control	Add product which user click buy to cart. Allow user to increase/decrease, remove the amount of product from cart.

7	Search product	Allow customers to search a specific product on shopping page.
8	Filter product	Allow user to sort product by brands.
9	Show product detail	Display product information.

4. Expected interface:

Table 2 Interface

Page	Purpose
Homepage	Shop homepage.
About page	Contains shop information.
Shopping page	List all product available
Check out page	Cart and costumer's information for payment.
Admin page	Allow administrator to manage products.
Login/Logout, Register page	Allow user to sign in, sign out and register to the website.
Buy history page	Display all product that user has bought.

II. Work distribution:

Table 3 Work Distribution

ID	Name	Work detail	Contribution
20110352	Nguyễn Việt Anh	<ul style="list-style-type: none"> Connect to database: <ul style="list-style-type: none"> Load data: list of products. Search product. Filter product. 	100%

		<ul style="list-style-type: none"> ○ User buy history. • Send email: verify code, bill confirmation. • Admin: add, delete, edit product. • Front-end customize. 	
20110403	Lê Y Thiện	<ul style="list-style-type: none"> • Login/Logout, Register using session/cookie. • Cart control using session/cookie. <ul style="list-style-type: none"> ○ Add, remove product to cart. ○ Process amount of product in cart. • Store invoice to database. • Front-end customize. 	100%

III. Design

1. Algorithms:

Our website build base on Model–view–controller (MVC) architecture. That divide the related program logic into three interconnected elements. This is done to separate internal representations of information from the ways information is presented to and accepted from the user.

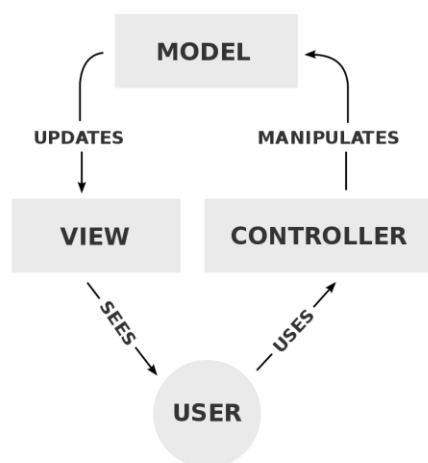


Figure 1 MVC model

We use jdbc to connect to database use Clear Database MySQL.

Process request: post and get method.

2. Classes:

A. All classes

Author: Nguyễn Việt Anh 20110352 + Lê Y Thiện 20110403

Table 4 Author's class: Thien & Anh

Order	Class name	Purpose
1	ProductControl (extends HttpServlet)	Annotation: /productload Load all product to website.
2	LoadData	Execute query to load/store data from database for different other function

Table 5 Author's class: Nguyen Viet Anh

Order	Class name	Purpose
1	DataBaseConnection	Connect to database.
2	User	Create User object to store user information and used further function.
3	Product	Create Product object to store user information and used further function.
4	Type	Create Type object to store type of object for filter function.
5	SendEmail	Send email: send code for verification and send bill for order confirmation.
6	AdminControl (extends HttpServlet)	Annotation: /admincontrol Load the list of products in the database. Applying paging function.
7	AddControl (extends HttpServlet)	Annotation: /add Add a new product to database.
8	DeleteControl (extends HttpServlet)	Annotation: /adminDetail Delete a product by ID from database.
9	AdminDetailControl (extends HttpServlet)	Annotation: Load the product information by ID and direct to edit.jsp.
10	AdminEditControl (extends HttpServlet)	Annotation: /edit Get the product information from edit.jsp page then edit product in database.
11	EmailVerifyControl (extends HttpServlet)	Annotation: /delete Send a verification code to user email after filled in register.jsp.

12	VerifyCodeControl (extends HttpServlet)	Annotation: /verifycode Check the verification code user input correct
13	ListProduct (extends HttpServlet)	Annotation: /ListProduct Load Product to different pages with index.
14	CatergoryControl (extends HttpServlet)	Annotation: /category Display all products of a specific brand.
15	DetailControl (extends HttpServlet)	Annotation: /detail Display all information of a specific product.
16	SearchControl (extends HttpServlet)	Annotation: /search Display all product with the name contains character user inputs at search bar.
17	UserHistoryControl (extends HttpServlet)	Annotation: /userhistorycontrol Display all product of user has bought.

Author: Lê Y Thiện 20110403

Table 6 Author's Class: Le Y Thien

Order	Class name	Purpose
1	LineItem	Create an object to store Product and the amount which was bought by user.
2	Cart	Create an object to store the list of LineItem which was bought buy customer
3	BeforeLogin (extends HttpServlet)	Annotation: /beforelogin Check the account if an account has been logged in recently
4	LoginControl (extends HttpServlet)	Annotation: Check the user if an account was registered or not
5	LogoutControl	Annotation: /login

	(extends HttpServlet)	Exit current account
6	RegisterControl (extends HttpServlet)	Annotation: /logout Support for register an account
7	BuyControl (extends HttpServlet)	Annotation: /buy Support for buying items from website then add to cart
8	ShowCartControl (extends HttpServlet)	Annotation: /checkout Display a list of products and quantities purchased by customer
9	ProcessAmount (extends HttpServlet)	Annotation: /process Supporting adjust quantities of Products purchased and remove it
10	Payment (extends HttpServlet)	Annotation: /payment Store the bill to database and send email to confirm bill

B. All Methods:

Author: Nguyễn Việt Anh 20110352

Table 7 Author's Method: Nguyen Viet Anh

Order	Method	Purpose	FileName (Line Contain Declaration)
1	processRequest(request, response) Input: get “index” at admin.jsp from request. Output: Push to request: Index as “ activeIndex”. endPage as “endP”.	<ul style="list-style-type: none"> • Load all product from database and apply Pagination. Each page will have 9 products. • Load the all the type of product. • 	AdminControl.java (35)

	listProductIndex as “adminListP”. list as “adminListT” Direct request to page: admin.jsp		
2	processRequest(request, response) Input: get all product’s information from request. Output: redirect request to admincontrol servlet	<ul style="list-style-type: none"> • Get id, name, image1, image2, image3, price, size. description, type from request • Insert to database a new product with the information above. 	AddControl.java (32)
3	ProcessRequest(request, response) Input: get the “delid” from request. Output: redirect request to admincontrol servlet	<ul style="list-style-type: none"> • Get delid from request and conduct delete product which has that delid. 	DeleteControl.java (32)
4	ProcessRequest(request, response) Input: get the “pid” from request. Output: Push to request product p as “detail”, list of types list as “ListT” and send request to Edit.jsp	<ul style="list-style-type: none"> • Get the information of product has pid get from request and also get list of types. 	AdminDetailControl (35)
5	ProcessRequest(request, response)	<ul style="list-style-type: none"> • Edit product with the same id in database 	AdminEditControl (32)

	<p>Input:</p> <p>Get all product information from request</p> <p>Output: redirect to admincontrol servlet</p>	with the information get from request.	
6	<p>processRequest(request, response)</p> <p>Input: Get user information from request.</p> <p>Output:</p> <p>Push user to session and redirect to verify.jsp</p>	<ul style="list-style-type: none"> • Get user information from request. • Generate a verification code • Create a new user with information and code above. • Send the code to the user email. • Push user to session 	EmailVerifyControl.java (33)
7	<p>processRequest(request, response)</p> <p>Input: get user from session</p> <p>Output:</p> <p>Direct to verify.jsp if code is incorrect.</p> <p>Direct to register servlet if code is correct.</p>	<ul style="list-style-type: none"> • Get the user from session • Get authcode from request. • Compare authcode with user.getCode() 	VerifyCodeControl.java (33)
8	<p>processRequest(request, response)</p> <p>Input:</p> <p>Output:</p>	<ul style="list-style-type: none"> • Calculate ending page for Pagination. 	ListProduct.java (32)

	Push to request endPage as “endP” and direct to shop.jsp		
9	<p>processRequest(request, response)</p> <p>Input: Get all type from request.</p> <p>Output: Push to request: list of products by type: list as “ListP” list of types: list as “ListT” 3 latest products: last3P as “Last3p” And direct to shop.jsp</p>	<ul style="list-style-type: none"> • Load from database all product with type get from request. 	CategoryControl.java (35)
10	<p>processRequest(request, response)</p> <p>Input: get pid from request</p> <p>Output: push to request product: p as “detail” and direct to single.jsp</p>	<ul style="list-style-type: none"> • Load product by ID get from request. 	DetailControl.java (33)
11	<p>processRequest(request, response)</p> <p>Input: get search from request</p> <p>Output: Push to request: List of products by name: list as “ListP” list of types: list as “ListT”</p>	<ul style="list-style-type: none"> • Load product by name base on search bar. 	SearchControl.java (35)

	3 latest products: last3P as “Last3p” And direct to shop.jsp		
12	processRequest(request, response) Input: get userid from request Output: Push to request list of product that user has bought: list as “ListUH” and direct to buyhistory.jsp	<ul style="list-style-type: none"> • Load the list of product has bought of the user has userid get from request. 	UserHistoryControl.java (34)
13	sendCode(User user) Input: user Output: true/false	<ul style="list-style-type: none"> • Send a email contain verify code to user.getEmail() 	SendEmail.java (33)
14	sendBill(User user, String fullName, String phoneNumber, String address, String total) Input: user, fullName, phoneNumber, address, total. Output: true/false	<ul style="list-style-type: none"> • Send a bill to customer email to customer email contain: customer name, phonenumber, address and total money of that bill. 	SendEmail.java (72)

Table 8 Author's Method: Le Y Thien

Order	Method	Purpose	FileName (Line Contain Declaration)
1	doPost (request, response) Input: get user and password from request. Output: Set username and password into request and redirect to LoginPage	Check the account if an account has been logged in recently	BeforeLogin.java (23)
2	doPost (request, response) Input: get Username and Password from request Output: Set an account to request and cookie and redirect to HomePage	+Check the account in database if an account has been registered or not. +Check this account is customer or admin	LoginControl.java (23)
3	doPost (request, response) Input: do not have Output: Remove account from session/cookie and remove Cart of this account. Redirect to HomePage	+Exit current account +Renew cart on website +Refresh cookie	LogoutControl.java (27)
4	doPost (request, response) Input: get information from form in View	+Supporting user create an account for shopping	RegisterControl.java (21)

	<p>Output: Store account into database and set an account to session/cookie.</p> <p>Redirect to HomePage</p>		
5	<p>doPost (request, response)</p> <p>Input: get Amount and Id of Product</p> <p>Output: Store Product to Cart then set Cart into cookie.</p>	+Supporting buy items by customer	BuyControl.java (27)
6	<p>doPost (request, response)</p> <p>Input: get list of products from cookie and store into Cart</p> <p>Output: Set Cart into request.</p> <p>Redirect to CheckoutPage</p>	<p>+Display the list of products which was bought by customer</p> <p>+Show total money of customer's Cart</p>	ShowCartControl (26)
7	<p>doGet (request, response)</p> <p>Input: get product Id and signal increase or decrease product.</p> <p>Output: Set Cart into cookie</p> <p>+Refresh current page</p>	+Supporting the change of product quantity	ProcessAmount.java (80)
8	<p>doPost (request, response)</p> <p>Input: get product Id</p> <p>Output: Set new Cart into cookie</p> <p>+Redirect to CheckoutPage</p>	+Supporting for remove products	ProcessAmount.java (36)

3. Database Desgin

Table Name	Fields
gMUWtf0kJX Invoice	ID_In : int(5) Date : date UID : varchar(10) TotalMoney : double Phone : varchar(20) Address : varchar(500)
gMUWtf0kJX User	ID : int(5) Name : varchar(500) Email : varchar(100) Username : varchar(10) Password : varchar(10) perMiss : int(11)
gMUWtf0kJX Product	ID_P : varchar(5) Name_P : varchar(20) Price : int(10) Type : varchar(10) Size : int(3) Image1 : varchar(500) Image2 : varchar(500) Image3 : varchar(500) Description : varchar(1500)
gMUWtf0kJX Detail_In	ID_In : int(5) ID_P : varchar(5) Amount : int(5)

Figure 2 Database

Table 9 List of table of database

Order	Table name	Purpose
1	User	Store user information (name, email, username, password, permission)
2	Product	Store product information (ID, name, price, type, size, images, descriptoin)
3	Invoice	Store Invoice of a user (ID, date, userID, total money, Phone, Address)
4	Detail_In	Store detail of a Invoice (ID of invoice, ID of product, amount)

4. Fields of each table:

Field of User table:

Table 10 Field of User

Order	Column name	Data type	Purpose
1	ID	Int	Store user's ID, auto increase
2	Name	varchar	Store user's name
3	Email	varchar	Store user's email
4	Username	varchar	Store user's username
5	Password	varchar	Store user's password
6	perMiss	int	Store user permission (admin or not)

Field of Product table:

Table 11 Field of Product

Order	Column name	Data type	Purpose
1	ID_P	Varchar	Store product ID
2	Name_P	varchar	Store product name
3	Price	varchar	Store product price
4	Type	Varchar	Store product type
5	Size	Int	Store product size
6	Image1	Varchar	Store product first image
7	Image2	Varchar	Store product second image
8	Image3	Varchar	Store product third image
9	Description	Varchar	Store product description

Field of Invoice table:

Table 12 Field of Invoice

Order	Column name	Data type	Purpose
1	ID_In	Int	Store invoice ID
2	Date	date	Store date of purchasing.
3	UID	varchar	Store user ID who purchased.
4	TotalMoney	Double	Store the total money
5	Phone	varchar	Store user phone
6	Address	Varchar	Store user address

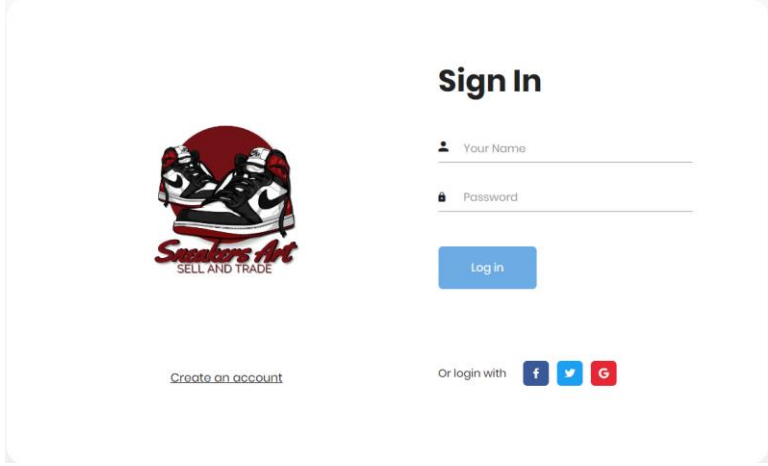
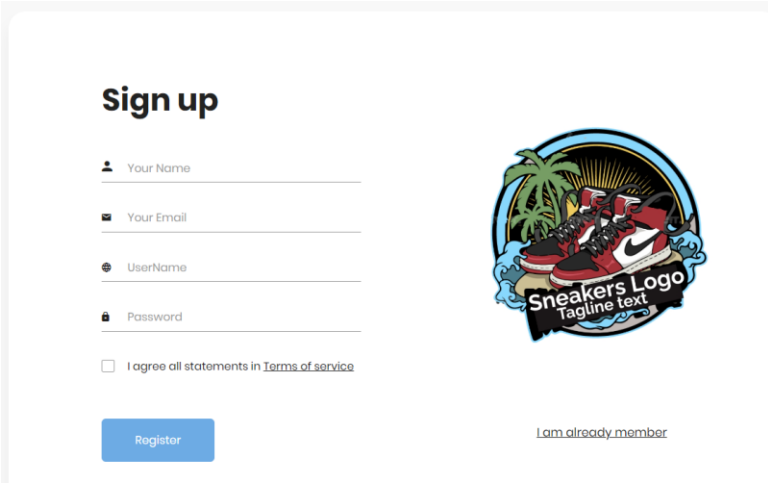
Field of Detail_In table:


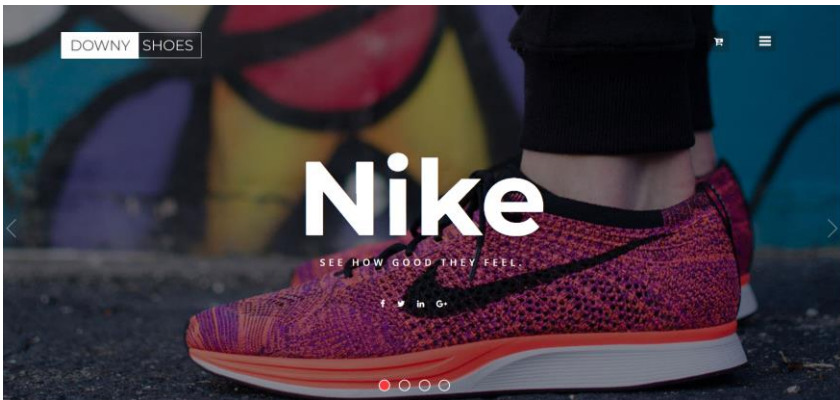
Table 13 Field of Detail_In

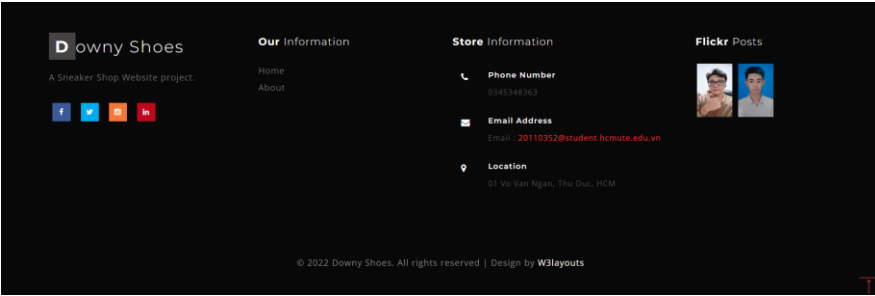
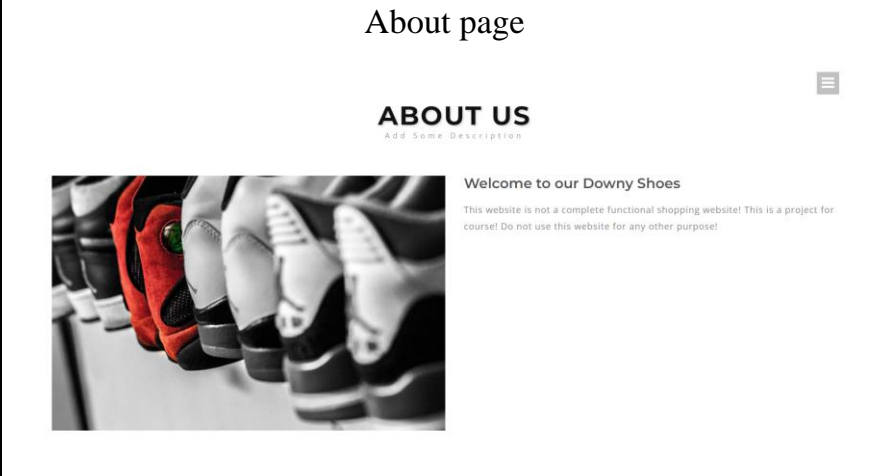
Order	Column name	Data type	Purpose
1	ID_In	Int	Store the ID of Invoice
2	ID_P	varchar	Store the product ID
3	Amount	varchar	Store the amount of a product user bought





5. Interface design

Table 14 Interface Design

	Name	Purpose	Explain
1	<p>Login page</p> 	Login	A page for user to login with username and password.
2	<p>Sign up page</p> 	Sign up	A page for new user to register with name, username, password, email.
3	Verify code	Verify code send to user email	User input the code which has been send to their email to verify account.






	<div> <div> <h1>Verify your email</h1> <div> <div> <div>🔒</div> <div>Your code</div> </div> <div> <input type="text"/> </div> </div> <div> <div>Verify</div> </div> </div> <div>  </div> </div>		
4	<div>Home page</div> 	introduction	Contain shop information, introduction
5	<div>Navigation bar</div> <div> <div>Home</div> <div>About</div> <div>Admin</div> <div>Buy History</div> <div>Shop Now</div> <div>Hello: an1@gmail.com</div> <div>Logout</div> </div>	A navigation bar	Always show as a button to trigger at every page. Allow user to choose different action such as: <ul style="list-style-type: none"> About: see the shop information Admin: move to admin to manage products

			<ul style="list-style-type: none"> • Buy history: move to user's buy history page • Shop now: move to shopping page • Logout: Sign out
6	<p>Footer</p> 	Show information	<p>Contain contact and shop information.</p> <p>And always display at the bottom of every page.</p>
7	<p>About page</p> 	Detail information of website	Contain detail information of the website
8	Admin page	Admin page	<p>A page for admin to manage list of products available.</p> <p>Admin can add or delete</p> <p>Edit button will direct to edit page which contain all information of selected product.</p>

	<div><div>Manage Product</div><div><div><div><div><div><div></div><div>ID</div></div><div><div>Name</div></div><div><div>Image</div></div><div><div>Price</div></div><div><div>Actions</div></div></div></div><div><div><div><div><div></div><div>sp001</div></div><div>superstar</div><div></div><div>2500000</div><div><div><div></div><div></div></div></div></div><div><div><div><div><div></div><div>sp002</div></div><div>stan smith</div><div></div><div>2500000</div><div><div><div></div><div></div></div></div></div><div><div><div><div><div></div><div>sp003</div></div><div>ultra 4dfwd</div><div></div><div>6000000</div><div><div><div></div><div></div></div></div></div><div><div><div><div><div></div><div>sp004</div></div><div>nmd r1</div><div></div><div>3600000</div><div><div><div></div><div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div>		
9	<div><div>Edit page</div><div><div>Edit Product</div><div><div><div><div><div><div></div><div>Edit Product</div><div></div></div><div><div><div>ID</div><div>sp001</div></div><div><div>Name</div><div>superstar</div></div><div><div>Image 1</div><div>imageprod/sp001/sp001_superstar.jpg</div></div><div><div>Image 2</div><div>imageprod/sp001/sp001_superstar_1.jpg</div></div><div><div>Image 3</div><div>imageprod/sp001/sp001_superstar_2.jpg</div></div><div><div>Price</div><div>2500000</div></div><div><div>Size</div><div>39</div></div><div><div>Description</div><div>Originally designed for the basketball court in the '70s. Revered by hip hop stars in the '80s. The adidas Superstar shoe has now become a streetwear</div></div><div><div>Category</div><div>adidas</div></div><div><div>Edit</div></div></div></div></div></div></div></div></div>	Allow admin to edit product.	Contain all information of selected product. Allow admin to adjust the information (except the ID)
10	<div><div>Buy history page</div></div>	List all products user has bought	List all products user has bought so that customers can review the bill or control their buying.

11

Buy History

<input type="checkbox"/>	ID	Name	Image	Price
<input type="checkbox"/>	sp001	superstar		2500000
<input type="checkbox"/>	sp002	stan smith		2500000
<input type="checkbox"/>	sp005	air jordan low		2929000
<input type="checkbox"/>	sp023	tennis dmsh 102		853000
<input type="checkbox"/>	sp006	pg 6 ep		3239000

Shopping page

Search Here...


Product name...

Brand

nike


adidas

Latest Product




sport dm 076

519000



sport dm 075


731000




sport dmsh 097


1236000

40% off



50% off






SUPERSTAR

2500000

★★★★☆

BUY




STAN SMITH

2500000

★★★★☆

BUY





ULTRA 4DFWD


6000000

★★★★☆

BUY



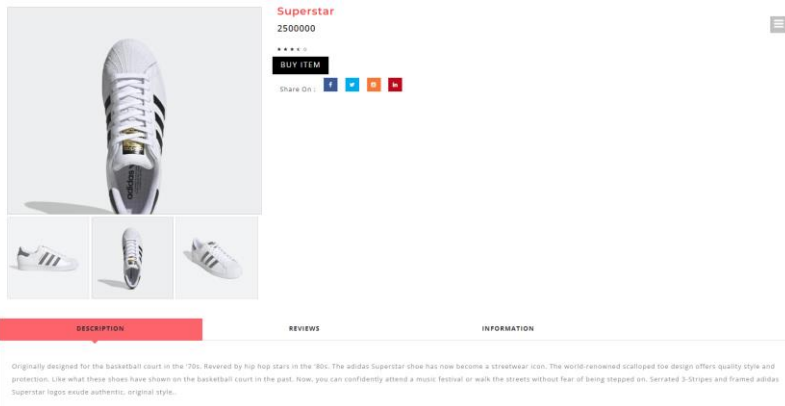




- Search bar: allow customers to search product by name
- Brands filter: allow customers to filter products by brands.
- Latest products: show 3 last product of the store.
- Buy button: add the selected product to cart.
- Click at the image of product: direct to the detail page of selected product.

12

Product detail page:

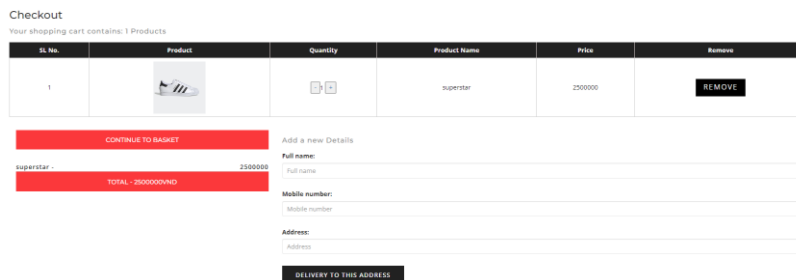


Show the detail of selected product.

- Show different images of product
- Show name and price
- Buy items button: add product to cart
- Description: Some other feature of product.

13

Checkout page



Show cart and check out

- Show the list of product user has selected
- Allow user to increase/decrease the quantity or remove entire product.
- Show the total money
- Allow customers to enter information: name, address for delivery, phone number.
- Delivery button: generate invoice and store to database, send a confirm bill to customer address.

14

Confirm bill mail:



Hoa Don Thanh Toan
Thu Duc, Viet Nam, 84
Phone: 0345348363 | [Email: 20110352@student.hcmute.edu.vn](mailto:20110352@student.hcmute.edu.vn)
Order Date: 2022-12-07

Delivery Addresss

So 1, Vo Van Ngan
Phuong Linh Chieu
Thanh Pho Thu Duc, Viet Nam

Billing Address

dia chi
Viet Nam

Detail Receipt

Name: Nguyễn Việt Anh
Email: vietanhnguyen.071002@gmail.com
Address: TP HCM

Phone: 0345348363
Date: 2022-12-07

Price Ship: 50\$

Total:
8668000.0

Note: Thank You For Purchasing From Us

An email layout sends to customer to confirm the bill.








- An email layout sends to customer to confirm the bill.
- Includes: Customer name, phone number, address, date, total money for customer services.

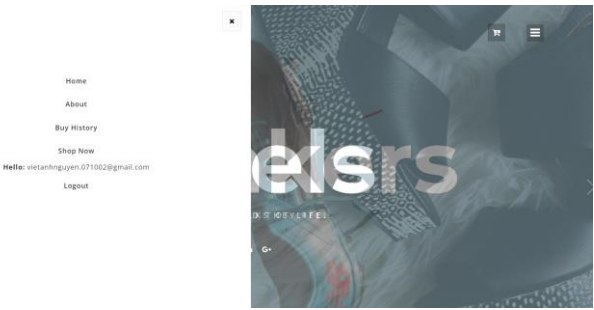
IV. Deploying and Testing:

Scenario 1: Signup/Log in before conduct buying.

User do not have account => Sign up => Verify code => Shopping.

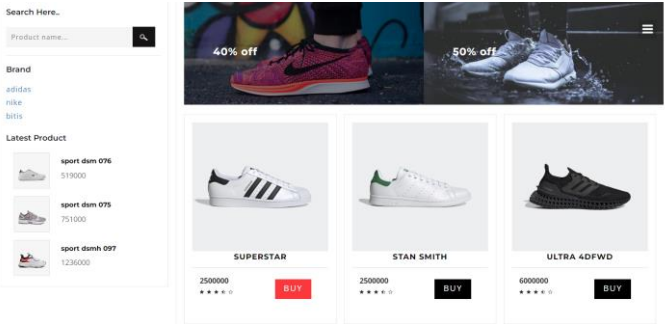
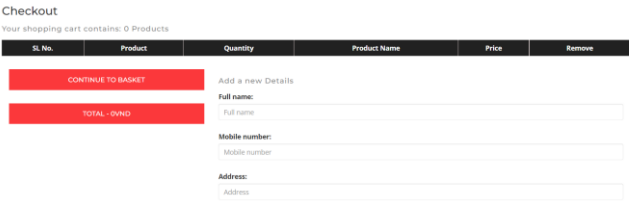
Table 15 Scenario 1

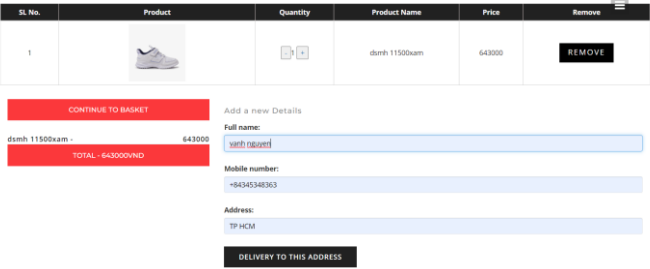
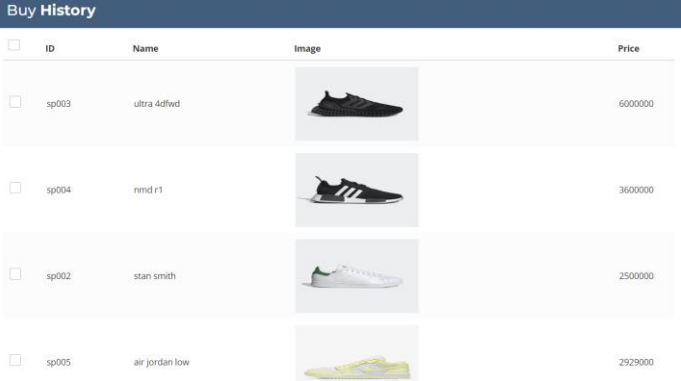
	Stage	Purpose	Explain
1	<p>Sign up</p> <p>Input: customer information</p> <p>Output: direct to verify page</p> <div> <div> <h3>Sign up</h3> <div> <div> vanh nguyen</div> <div> vietanhhnguyen.071002@gmail.com</div> <div> 123</div> <div> ...</div> <div><input checked="" type="checkbox"/> I agree all statements in Terms of service</div> <div>Register</div> </div> <div>  <p>I am already member</p> </div> </div> </div>	Register	Get User information and send verification code to confirm account.
2	<p>Verify code</p> <p>Input: user get the code at email</p> <p>Output:</p> <p>Redirect to verify page if the code is wrong.</p> <p>Move to homepage if the code is valid</p> <div> <div> <h3>Verify your email</h3> <div> <div> Your code</div> <div>Verify</div> </div> <div>  </div> </div> </div>	Verify	Check the code user input.

3	<p>Homepage:</p> 	Logged in	As can be seen at the navigation bar. User has successfully sign up and login with email show as: hello + “user email”
---	--	-----------	--

Scenario 2: Conduct shopping
 User choose product => add to cart => check out

Table 16 Scenario 2

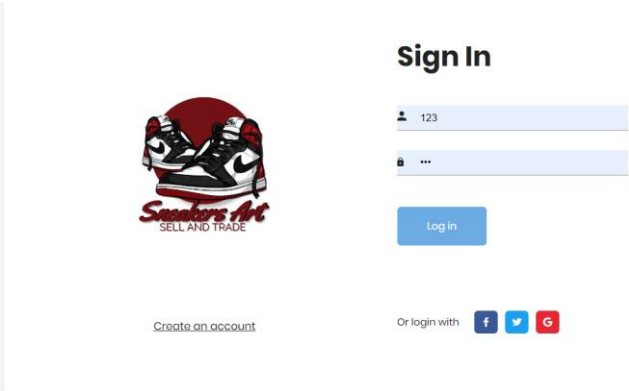
	Stage	Purpose	Explain
1	<p>Select product and add to cart</p> <p>Input: Product id</p> <p>Output: direct to verify page</p> 	Shopping, add product to cart	<p>If the buy button of product is clicked.</p> <p>That product will be added to cart with quantity = 1.</p>
2	<p>Checkout if the cart is empty</p> 	Check out	<p>If the cart is empty then the checkout button will not appear.</p>

3	<p>Check out if the cart is not empty</p> <p>Input: user name, address, phone number</p> 	Check out	User have to enter information for delivery and confirm bill.
4	<p>Check the buy history</p> 	Review bill	User can check the list of buying product.

Scenario 3: Conduct shopping without signed in
User choose product => add to cart => check out => sign in/sign up

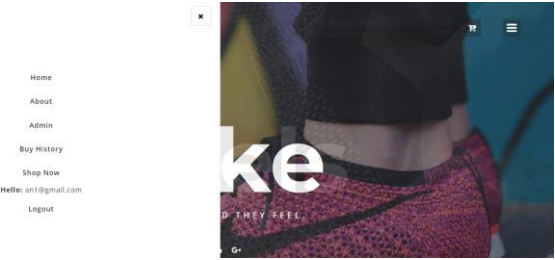
Table 17 Scenario 3

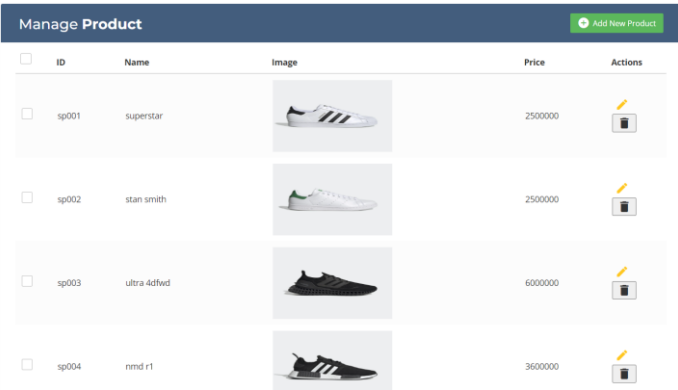
	Stage	Purpose	Explain
1	Check out page (when user not logged in)	Check out	If the user not logged in and click delivery button then the web will direct to sign in page

			
2	<p>Sign in</p> 	Sign	Require user to sign in to conduct check out.

Scenario 4: Admin sign in to
User choose product => add to cart => check out => sign in/sign up

Table 18 Scenario 4

	Stage	Purpose	Explain
1	<p>User login with admin account:</p> 	Login as admin	A admin button will appear on the navigation bar, direct to admin page

2	<p>Admin page</p> 	Manage products	Admin can add, delete or edit product
---	--	-----------------	---------------------------------------

V. Conclusion

Pros:

Through the working progress, our group has learned and practice all of the subject which we have learn from the beginning to successfully build a shopping website.

The web after being deployed and tested is working properly, functionally. The speed response of the web is fast and productive.

All member has completed their assignment and do not have any conflict.

Cons:

We used the template online.

The website is only had simple function and do not have much feature.

The database structure is uncomplicated.

Future development:

We will re design the front-end of website to make it more beauty and add more function, feature so that the website can be use in business someday.

Link source code:

https://github.com/NVA071002/TT_Project.git

Our website has been deploy to Heroku:

<https://itprojectvanhthien.herokuapp.com>

Reference Source:

Template of the website: <https://w3layouts.com>

Template of login/register : <https://colorlib.com>