VIETNAM INTERNATIONAL UNIVERSITY – HO CHI MINH CITY

INTERNATIONAL UNIVERSITY



WEB APPLICATION DEVELOPMENT PROJECT

Ecommerce Project

**Members of Group:**

Lê Thành Đại Dương\_ITITUN14002

Vương Khắc Duy\_ITITUN14010

Ninh Thế Lâm\_ ITITUN14007

GitHub Link: <https://github.com/tieuduong241096/2017S2-WEB_LAB->

Outline

1. Introduction
   1. Overview

Our web site base on Ecommerce web application providing high standards in functionality, precision and convenient. Our target markets are all people who need purchase online.

* 1. Motivation

Starting from Ecommerce project in the university course called “Web Application Development”, the team need to improve this website with many new features which make the website more professional. brainstorms suggested many fascinating and extraordinary ideas: from many famous website for example “Amazone”, “Alibaba”, “Ebay” and all of them have common features such as add product into cart, payment product, view the history purchase, vote for product, recommend, search, etc. Finally, our team decide adding four functions is that view the history purchase, recommend, search and comment by facebook.

In the future, we will continue to update the website with more attractive new features, and more convenient for users.

1. Organization and Roles

Our team includes three members. Each of us have separate missions, responsibilities, and working roles which affect the progress of the project. The roles and every contact information for each members are shown in the following table:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Role** | **Contact** | **Email Address** |
| Le Thanh Dai Duong | Leader  Recommend  Comment  Documentation |  |  |
| Ninh The Lam | Developer  Search  Documentation |  |  |
| Vuong Khac Duy | Developer  Purchase History  Documentation |  |  |

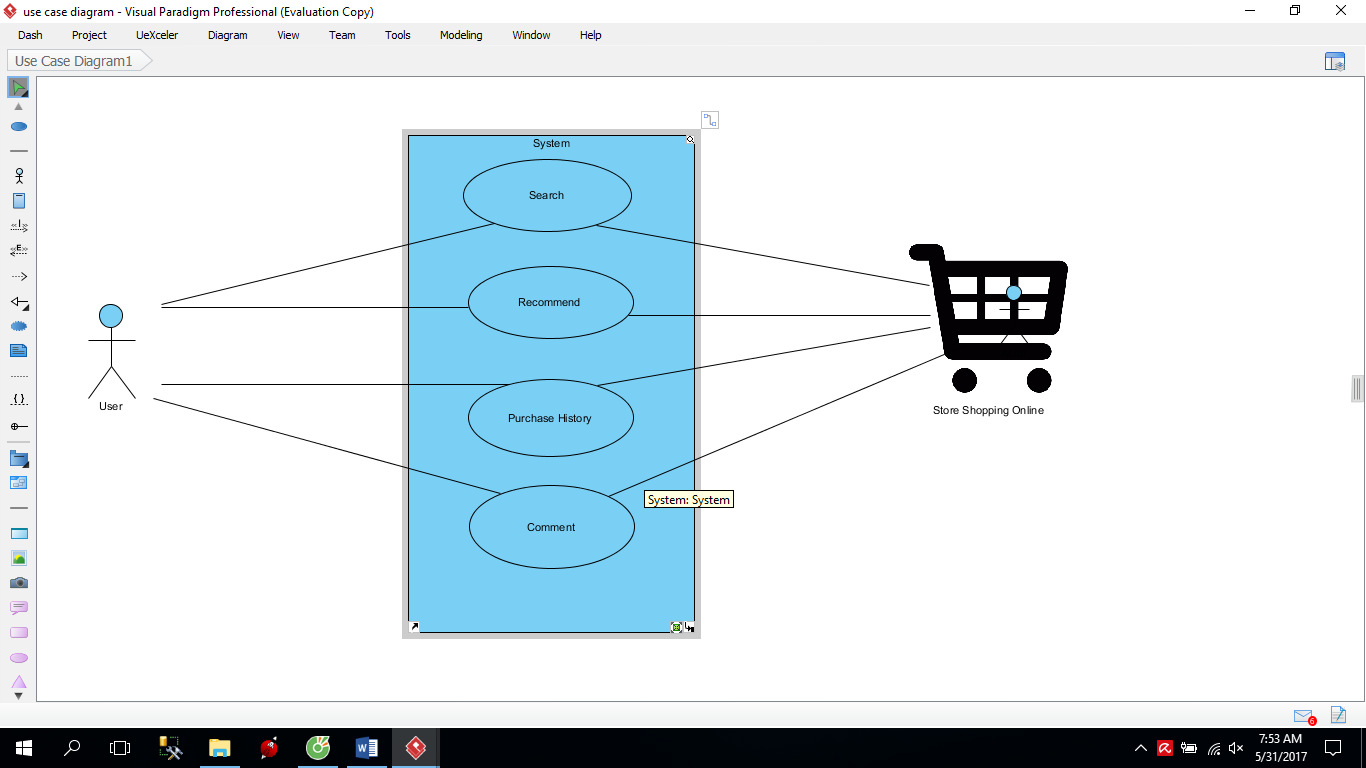
If you have any question relating to the project and its elements, feel free to contact us.

1. Progress Approximation

|  |  |
| --- | --- |
| Time Delivery | The first day of the construction: 20th March 2017  The last day of the construction: 27th May 2017  Duration : 2 month  Demonstration: 2nd June 2017 |
|  |  |

**User Analysis**

Use Case Diagrams.



**Use Case Description**

## USE CASE 1:

**Name**: Search Product

**Description:** The list of products is showed when the user input the name of products which they want to find.

**Identifier** UC1

**Inputs**: Input desired product name

**Output**: Show all products that have the letters that user typed in

**Basic Course**

|  |  |
| --- | --- |
| ACTOR(User) | SYSTEM |
| 1.Input Product name | Display the all of products |
| 2.Click on The product | Display the photo and name of product |

**Precondition:** None

**Post condition:** None

**User Story:** As a customer, I want to search the products.

## USE CASE 2:

**Name**: Purchase History

**Description:** Show the list of purchase information of the user

**Identifier**: UC2

**Inputs**: User ID , Password

**Output**: Information of purchase history

**Basic Course**

|  |  |
| --- | --- |
| ACTOR(User) | SYSTEM |
| 1. Login | Display login page |
| 1. Input User Name and Password |  |
| 1. Submit | * Check username and password * If true, show the purchase history * Else return to Login |

**Precondition:** Users have to log in with their own account.

**Post condition:** None.

**User story:** Customers can check the purchase history in the future.

## USE CASE 3:

**Name**: Recommend Items

**Description:** Show a several items which are in top purchase.

**Identifier** UC3

**Inputs**: none

**Output**: The list of recommend product based on mutual friends interested

**Basic Course**

|  |  |
| --- | --- |
| ACTOR(User) | SYSTEM |
| 1. Access to home page | * Show the list of recommend products. |

**Precondition:** None

**Post condition:** None

**User story:** The list of recommend products.

## USE CASE 4:

**Name**: Comment Items

**Description:** Show a several comment which are below purchase.

**Identifier** UC4

**Inputs**: User must have available account.

**Output**: The list of comment products .

**Basic Course**

|  |  |
| --- | --- |
| ACTOR(User) | SYSTEM |
| 1. User “click” on the random product. 2. Input the normal text. 3. User “Enter” | * 1. The system displays the available comment and response.   2.1 Display the content in the textbox .  3.1 The system displays user comment on the screen and save it in database. |

**Precondition:** None

**Post condition:** None

**User story:** The list of comment products.

**Design**

**System Architecture MVC (Model - View - Controller) pattern.**

**View**

Main Interface

+header.jsp

+index.jsp

User Interface

+account.jsp

+cart.jsp

+menu.jsp

+searchContent.jsp

+feedbacl.jsp

+recommendItem.jsp

Desigm Framework

+folder : css

+folder :bootstrap

+folder : images

**Controller**

Package:Controller

AccountController.java

Cartservlet.java

CheckoutController.java

FeedbackController.java

LoginServlet.java

SearchServlet.java

Form to display

User events

Change notification **Change notification**

Update request

Model

Package :Connect

+DBConnect.java

Package :model

+Cart.java

+Category.java

+Feedbacl.java

+Product.java

Receipt.java

+ReciptDetail.java

+Sizebean.java

+User.java

+ReciptReceiptDetailProduct.java

Refresh request

**Class Diagram**

|  |
| --- |
| **Product** |
| productID: Integer  productName: String  productPrice: Double  productImage: String  productQuantity: Integer  description: String  discount: Double  count: Integer |
| getProductID(): Integer  setProductID(productID : Integer): void  getProductName(): String  setProductName( productName: String): void  getProductPrice(): Double  setProductPrice( productPrice: Double ): void  getProductImage(): String  setProductImage( productImage: String): void  getProductQuangtity(): Integer  setProductQuantity( productQuantity: Integer): void  getDescription(): String  setDescrption( description: String): void  getDiscount(): Double  setDiscount( discount: Double): void  getCount(): Integer  setCount( countNumber: Integer): void |

|  |
| --- |
| **Search** |
| productID: Integer  productName: String  productPrice: Double  productImage: String  productQuantity: Integer  description: String  discount: Double  count: Integer |
| getProductID(): Integer  setProductID(productID : Integer): void  getProductName(): String  setProductName( productName: String): void  getProductPrice(): Double  setProductPrice( productPrice: Double ): void  getProductImage(): String  setProductImage( productImage: String): void  getProductQuangtity(): Integer  setProductQuantity( productQuantity: Integer): void  getDescription(): String  setDescrption( description: String): void  getDiscount(): Double  setDiscount( discount: Double): void  getCount(): Integer  setCount( countNumber: Integer ): void |

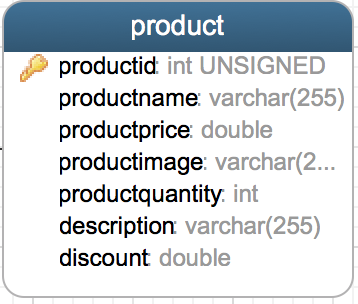
|  |
| --- |
| **Product** |
| ProductID: Integer  productName: String  productPrice: Double  productImage: String  productQuantity: Integer  description: String  discount: Double  count: Integer |
| getProductID(): Integer  setProductID(productID : Integer): void  getProductName(): String  setProductName( productName: String): void  getProductPrice(): Double  setProductPrice( productPrice: Double ): void  getProductImage(): String  setProductImage( productImage: String): void  getProductQuangtity(): Integer  setProductQuantity( productQuantity: Integer): void  getDescription(): String  setDescrption( description: String): void  getDiscount(): Double  setDiscount( discount: Double): void  getCount(): Integer  setCount( countNumber: Integer ): void |

5. Database Diagram

Entity Description

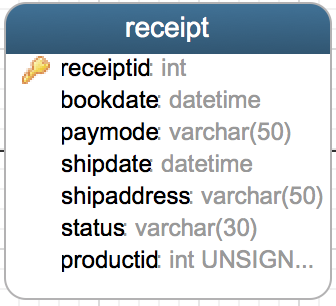
**PRODUCT**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Field name | Type (size) | Identity | NULL | PK | FK | References | Description |
| ProductID | int | yes | no | PK |  |  | Product identity number |
| ProductName | varchar(255) |  | no |  |  |  | Product Name |
| ProductPrice | double |  | no |  |  |  | Product’s Price |
| ProductImage | varchar(255) |  | no |  |  |  | Product’sImage |
| ProductQuantity | int |  | no |  |  |  | Product’s Quantity |
| Description | varchar(255) |  |  |  |  |  | Product’s Description |
| Discount | double |  |  |  |  |  | Product’s Discount |

****

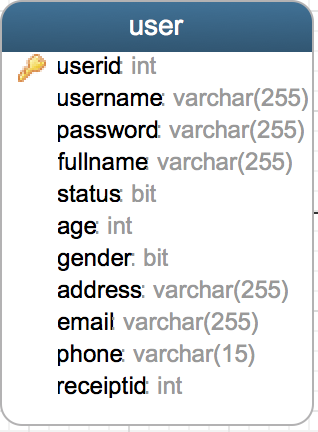
**PURCHASE HISTORY**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Field name | Type (size) | Identity | NULL | PK | FK | References | Description |
| ReceiptID | int | yes | no | PK |  |  | Receipt identity number |
| BookDate | datetime |  | no |  |  |  | User’s purchase date |
| Status | varchar |  | no |  |  |  | Status of receipt |
| ShipAddress | varchar |  | no |  |  |  | Address to arrive |
| PayMode | varchar |  | no |  |  |  | Pay mode of receipt |
| ShipDate | datetime |  | no |  |  |  | Ship Date of receipt |
| ProductID | int |  | no |  | FK | Product(ProductID) | One receipt detail has many product |

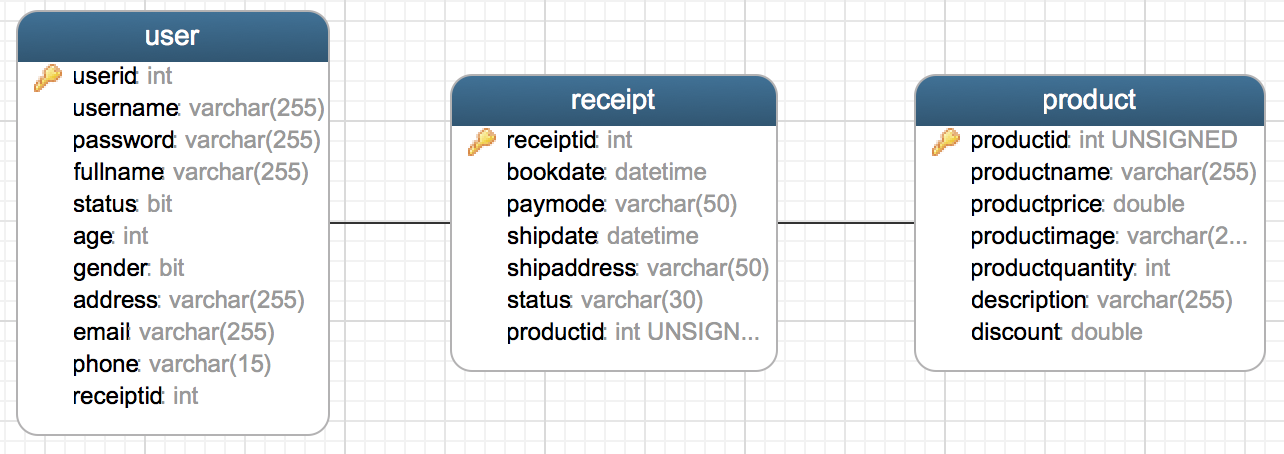
****

**USER**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Field name | Type (size) | Identity | NULL | PK | FK | References | Description |
| UserID | int | yes | no | PK |  |  | User identity number |
| UserName | varchar(30) |  | no |  |  |  | User name to login |
| Password | varchar(30) |  | no |  |  |  | User’s password to login |
| UserAge | int |  | no |  |  |  | User age |
| UserGender | bit |  | no |  |  |  | bit 1: Male; bit 0: Female |
| UserAddress | varchar(255) |  | no |  |  |  | User’s Address |
| Email | varchar(255) |  | no |  |  |  | User’s Email |
| FullName | varchar(255) |  | no |  |  |  | User’s FullName |
| Phone | varchar(15) |  | no |  |  |  | User’s Phone |
| ReceiptID | int |  | no |  | FK | Receipt(receipteid) | One user have one or more receipt |

****

Tables Design



Development Environment and Techniques

-HTML

-CSS

-Javascript

-Servlet

-Github :

Conclusion and Future Work