ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC BÁCH KHOA

KHOA KHOA HỌC & KỸ THUẬT MÁY TÍNH



**LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC**

**XÂY DỰNG WEBSITE BÁN HÀNG MÔ HÌNH C2C (CUSTOMER TO CUSTOMER)**

GVHD: ThS. ĐẶNG TRẦN TRÍ

----o0o----

SVTH1: HOÀNG THANH NGỌC BẢO - 50600117

SVTH2: BÙI ĐĂNG KHÁNH - 50601060

SVTH3: LÝ THĂNG LONG - 50601311

TP. HỒ CHÍ MINH, THÁNG 1 NĂM 2011

# Lời cảm ơn

Lời đầu tiên, em xin bày tỏ lòng biết ơn sâu sắc đến thầy Đặng Trần Trí, người đã trực tiếp hướng dẫn và tận tình giúp đỡ chúng em trong suốt quá trình thực hiện luận văn tốt nghiệp này.

Để có kết quả như ngày hôm nay, em cũng xin cám ơn tất cả Thầy, Cô trường Đại học Bách Khoa Tp. Hồ Chí Minh đã dạy dỗ, và truyền đạt những kiến thức quí báu trong suốt thời gian em theo học tập tại trường.

Xin kính chúc quý Thầy, Cô trường Đại học Bách Khoa Tp. Hồ Chí Minh sức khoẻ và thành công.

Và sau cùng, chúng con xin cảm ơn ba mẹ, những người đã sinh thành, dưỡng dục, nuôi dạy chúng con nên người, và đã luôn động viên chúng con trong suốt quá trình thực hiện luận văn tốt nghiệp này.

# MỤC LỤC

# DANH MỤC HÌNH

# DANH MỤC BẢNG

# CHƯƠNG 1. TỔNG QUAN VỀ THƯƠNG MẠI ĐIỆN TỬ & MÔ HÌNH C2C (Customer to Customer)

## 1.1 Thương Mại Điện Tử

Trong vài năm trở lại đây, đặc biệt là ở Việt Nam, cụm từ Thương Mại Điện Tử (TMĐT) (còn gọi là E-Commerce hay E-Business) xuất hiện ngày một phổ biến. Phạm vi của TMĐT rất rộng, bao quát hầu như mọi hình thái hoạt động kinh tế, không chỉ bao gồm buôn bán hàng hóa và dịch vụ, vì thế khó có thể tìm một định nghĩa có ranh giới rõ rệt cho khái niệm TMĐT. Xét một cách tổng quát, các định nghĩa TMĐT được chia thành hai nhóm tùy thuộc vào quan điểm:

* Theo nghĩa hẹp, TMĐT chỉ đơn thuần bó hẹp thương mại điện tử trong việc mua bán hàng hóa và dịch vụ thông qua các phương tiện điện tử, nhất là qua Internet và các mạng liên thông khác.
* Theo nghĩa rộng, TMĐT là các giao dịch tài chính và thương mại bằng phương tiện điện tử như: trao đổi dữ liệu điện tử, chuyển tiền điện tử và các hoạt động như gửi/rút tiền bằng thẻ tín dụng..

TMĐT có thể được phân loại theo tính cách của người tham gia:

* Người tiêu dùng
* C2C (Consumer-To-Comsumer) Người tiêu dùng với người tiêu dùng
* C2B (Consumer-To-Business) Người tiêu dùng với doanh nghiệp
* C2G (Consumer-To-Government) Người tiêu dùng với chính phủ
* Doanh nghiệp
* B2C (Business-To-Consumer) Doanh nghiệp với người tiêu dùng
* B2B (Business-To-Business) Doanh nghiệp với doanh nghiệp
* B2G (Business-To-Government) Doanh nghiệp với chính phủ
* B2E (Business-To-Employee) Doanh nghiệp với nhân viên
* Chính phủ
* G2C (Government-To-Consumer) Chính phủ với người tiêu dùng
* G2B (Government-To-Business) Chính phủ với doanh nghiệp
* G2G (Government-To-Government) Chính phủ với chính phủ

Người ta khai thác sức mạnh của TMĐT vì một số lý do như:

* TMĐT giúp cho các doanh nghiệp nắm được thông tin phong phú về thị trường và đối tác.
* TMĐT giúp giảm chi phí sản xuất.
* TMĐT giúp giảm chi phí bán hàng và tiếp thị.
* TMĐT thông qua Internet giúp người tiêu dùng và các doanh nghiệp giảm đáng kể thời gian và chí phí giao dịch.
* TMĐT tạo điều kiện cho việc thiết lập và củng cố mối quan hệ giữa các thành phần tham gia vào quá trình thương mại.
* TMĐT tạo điều kiện sớm tiếp cận nền kinh tế số hoá.

Với những lợi ích như vậy, TMĐT ngày càng phát triển mạnh mẽ, biểu hiện qua việc những trang kinh doanh trực tuyến xuất hiện ngày một nhiều trên Internet.

Các công đoạn tiêu biểu của một giao dịch trên các trang kinh doanh trực tuyến này:

1. Khách hàng, từ một máy tính tại một nơi nào đó, điền những thông tin thanh toán và điạ chỉ liên hệ vào đơn đặt hàng (Order Form) của Website bán hàng (còn gọi là Website TMĐT). Doanh nghiệp nhận được yêu cầu mua hàng hoá hay dịch vụ của khách hàng và phản hồi xác nhận tóm tắt lại những thông tin cần thiết nh mặt hàng đã chọn, địa chỉ giao nhận và số phiếu đặt hàng…
2. Khách hàng kiểm tra lại các thông tin và kích (click) vào nút (button) "đặt hàng", từ bàn phím hay chuột (mouse) của máy tính, để gởi thông tin trả về cho doanh nghiệp.
3. Doanh nghiệp nhận và lưu trữ thông tin đặt hàng đồng thời chuyển tiếp thông tin thanh toán (số thẻ tín dụng, ngày đáo hạn, chủ thẻ ...) đã được mã hoá đến máy chủ (Server, thiết bị xử lý dữ liệu) của Trung tâm cung cấp dịch vụ xử lý thẻ trên mạng Internet. Với quá trình mã hóa các thông tin thanh toán của khách hàng được bảo mật an toàn nhằm chống gian lận trong các giao dịch (chẳng hạn doanh nghiệp sẽ không biết được thông tin về thẻ tín dụng của khách hàng).
4. Khi Trung tâm Xử lý thẻ tín dụng nhận được thông tin thanh toán, sẽ giải mã thông tin và xử lý giao dịch đằng sau bức tường lửa (FireWall) và tách rời mạng Internet (off the Internet), nhằm mục đích bảo mật tuyệt đối cho các giao dịch thương mại, định dạng lại giao dịch và chuyển tiếp thông tin thanh toán đến ngân hàng của doanh nghiệp (Acquirer) theo một đường dây thuê bao riêng (một đường truyền số liệu riêng biệt).
5. Ngân hàng của doanh nghiệp gởi thông điệp điện tử yêu cầu thanh toán (authorization request) đến ngân hàng hoặc công ty cung cấp thẻ tín dụng của khách hàng (Issuer). Và tổ chức tài chính này sẽ phản hồi là đồng ý hoặc từ chối thanh toán đến trung tâm xử lý thẻ tín dụng trên mạng Internet.
6. Trung tâm xử lý thẻ tín dụng trên Internet sẽ tiếp tục chuyển tiếp những thông tin phản hồi trên đến doanh nghiệp, và tùy theo đó doanh nghiệp thông báo cho khách hàng được rõ là đơn đặt hàng sẽ được thực hiện hay không.

Toàn bộ thời gian thực hiện một giao dịch qua mạng từ bước 1 -> bước 6 được xử lý trong khoảng 15 - 20 giây.

## 1.2 Mô hình C2C

Tiêu biểu nhất của các trang kinh doanh trực tuyến hiện nay là các trang web thương mại bán hàng qua mạng Internet theo mô hình C2C.

Loại hình TMĐT này được phân loại bởi sự tăng trưởng của thị trường điện tử và đấu giá trên mạng, đặc biệt với các ngành theo trục dọc nơi các công ty/ doanh nghiệp có thể đấu thầu cho những cái họ muốn từ các nhà cung cấp khác nhau.

Loại hình TMĐT này tới theo ba dạng:

* Đấu giá trên một cổng, chẳng hạn như eBay, cho phép đấu giá trên mạng cho những mặt hàng được bán trên web.
* Hệ thống hai đầu như Napster (một giao thức chia sẻ dữ liệu giữa người dùng sử dụng diễn đàn nói chuyện IRC) và các hình thức trao đổi file và tiền.
* Quảng cáo phân loại tại một cổng như Excite Classifieds và eWanted (một thị trường mạng trao đổi qua lại nơi người mua và người bán có thể thương thuyết và với đặc thù “người mua hướng tới & muốn quảng cáo”).

Có rất ít thông tin về quy mô của thương mại điện tử C2C. Tuy nhiên, con số C2C về các trang web thông dụng C2C như là eBay và Napster chỉ ra rằng thị trường này thì rất lớn. Những trang web này tạo ra hàng triệu đô la bán hàng mỗi ngày.

Ở Việt Nam, hiện có một số website thương mại điện tử C2C lớn như Chợ Điện Tử, 123mua, 5s, Vatgia... Đặc biệt là eBay đã mở riêng 1 trang bằng tiếng Việt, điều đó cho thấy tiềm năng ở thị trường Việt Nam là rất lớn.

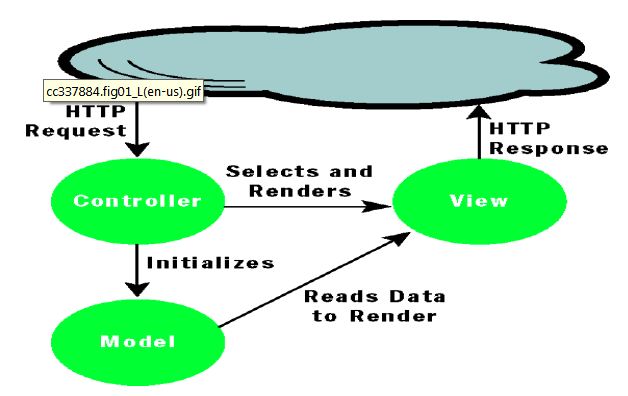
# CHƯƠNG 2. TỔNG QUAN VỀ ASP.NET MVC FRAMEWORK

## Khái niệm ASP.NET MVC

### Mô hình MVC cơ bản

MVC viết tắt của các chữ cái đầu của Models, Views, Controllers.

MVC chia giao diện UI thành 3 phần tương ứng: đầu vào của controller là các điều khiển thông qua HTTP request, model chứa các miền logic, view là những thứ được sinh ra trả về cho trình duyệt.



Hình 2.1 Mô hình MVC cơ bản

Lợi ích của việc dùng phương pháp MVC là sự phân đoạn rõ ràng giữa models, views, controllers bên trong ứng dụng. Cấu trúc sạch sẽ giúp cho việc kiểm tra lỗi ứng dụng trở nên dễ dàng hơn.

**Models** trong các ứng dụng dựa trên MVC là những thành phần có nhiệm vụ lưu trữ thông tin, trạng thái của các đối tượng, thông thường nó là một lớp được ánh xạ từ một bảng trong CSDL. Lấy ví dụ, chúng ta có lớp Product được sử dụng để mô tả dữ liệu từ bảng Products trong SQL, bao gồm ProductID, OrderDate…

Còn đối với **Views**, nó chính là các thành phần chịu trách nhiệm hiển thị các thông tin lên cho người dùng thông qua giao diện. Thông thường, các thông tin cần hiển thị được lấy từ thành phần Models. Ví dụ, đối tượng Product có một “Edit” view bao gồm các textboxes, các dropdowns và checkboxes để chỉnh sửa các thuộc tính của sản phẩm; có một “Display” view gồm 2 dòng, cột dòng là ProductID, dòng sau là OrderDate… để xem thông tin về sản phẩm.

Cuối cùng, **Controllers** trong các ứng dụng kiểu MVC chịu trách nhiệm xử lý các tác động về mặt giao diện, các thao tác đối với models, và cuối cùng là chọn một view thích hợp để hiển thị ra màn hình. Trong kiến trúc MVC, view chỉ có tác dụng hiển thị giao diện mà thôi, còn điều kiển dòng nhập xuất của người dùng vẫn do Controllers đảm trách.

### Một vài đặc trưng trong ASP.NET MVC

Tách rõ ràng các mối liên quan, mở khả năng test TDD (test driven developer). Có thể test unit trong ứng dụng mà không cần phải chạy Controllers cùng với tiến trình của ASP.NET và có thể dùng bất kỳ một unit testing framework nào như NUnit, MBUnit, MS Test, v.v…

Có khả năng mở rộng, mọi thứ trong MVC được thiết kế cho phép dễ dàng thay thế/tùy biến ( ví dụ: có thể lựa chọn sử dụng engine view riêng, routing policy, parameter serialization, v.v…). Bao gồm một ánh xạ URL mạnh mẽ cho phép xây dựng ứng dụng với những URL sạch, các URL không cần cs phần mở rộng (ví dụ: có thể ánh xạ địa chỉ /Products/Edit/4 để thực hiện hành động “Edit” của lớp điều khiển ProductControllers hoặc ánh xạ địa chỉ /Blog/SomeTopic để thực hiện hành động “Display Topic” của lớp điều khiển BlogEngineController).

ASP.NET MVC Framework cũng hỗ trợ những file ASP.NET như .ASPX .ASCX và .Master, đánh dấu các tập tin này như một “view template” ( có thể dễ dàng dùng các tính năng của ASP.NET như lồng các trang Master, <%= %> snippets, mô tả server controls, template, data-binding, localization, v.v… ). Tuy nhiên sẽ không còn postback và interactive back server và thay vào đó là interactive end-user tới một Controller class ( không còn viewstate, page lifecycle ).

ASP.NET MVC Framework hỗ trợ đầy đủ các tính năng bảo mật của ASP.NET như forms/windows authenticate, URL authorization, membership/roles, output và data caching, session/profile state, configuration system, provider architecture v.v…

## Sự khác biệt với Webfrom

ASP.NET WebForm sử dụng ViewState để quản lý, các trang ASP.NET đều có lifecycle, postback và dùng các web controls, các events để thực hiện các hành động cho UI khi có sự tương tác với người dùng nên hầu hết ASP.NET WebForm xử lý chậm.

ASP.NET MVC Framework chia ra thành 3 phần: Models, Views, Controllers. Mọi tương tác của người dùng với Views sẽ được thực hiện hành động trong Controllers, không còn postback, không còn lifecycle không còn events.

Việc kiểm tra ( test ), gỡ lỗi ( debug ) với ASP.NET đều phải chạy tất cả các tiến trình của ASP.NET và mọi sự thay đổi ID của bất kỳ controls nào cũng ảnh hưởng đến ứng dụng. Đối với ASP.NET MVC Framework thì việc có thể sử dụng các unit test có thể thẩm định rất dễ dàng các Controller thực hiện như thế nào.

|  |  |  |
| --- | --- | --- |
| **Tính năng** | **ASP.NET 2.0** | **ASP.NET MVC** |
| Kiến trúc chương trình | Kiến trúc mô hình WebForm -> Business -> Database | Kiến trúc sử dụng việc phân chia chương trình thành Controllers, Models, Views |
| Cú pháp chương trình | Sử dụng cú pháp của webform, tất các sự kiện và controls do server quản lý | Kiến trúc sử dụng việc phân chia chương trình thành Controllers, Models, Views |
| Truy cập dữ liệu | Sử dụng hầu hết các công nghệ truy cập dữ liệu trong ứng dụng | Phần lớn dùng LINQ to SQL class để tạo mô hình truy cập đối tượng |
| Debug | Debug chương trình phải thực hiện tất cả bao gồm các lớp truy cập dữ liệu, sự hiển thị, điều khiển các controls | Debug có thể sử dụng các unit test kiểm tra các phương thức trong controller |
| Tốc độ phân tải | Tốc độ phân tải chậm khi trong trang có quá nhiều các controls vì ViewState quá lớn | Phân tải nhanh hơn do không phải quản lý ViewState để quản lý các control trong trang |
| Tương tác với javascript | Tương tác với javascript khó khăn vì các controls được điều khiển bởi server | Tương tác với javascript dễ dàng vì các đối tượng không do server quản lý điều khiển không khó |
| URL Address | Cấu trúc địa chỉ URL có dạng <filename>.aspx?&<các tham số> | Cấu trúc địa chỉ rành mạch theo dạng Controllers/Action/Id |

Bảng 2.1 So sánh ASP.NET 2.0 và ASP.NET MVC

## Định tuyến URL và điều phối hiển thị

### Công dụng của hệ thống định tuyến trong ASP.NET MVC

ASP.NET MVC Framework có một hệ thống định tuyến URL ( URL Routing System) linh hoạt cho phép xác định các quy tắc ánh xạ địa chỉ URL bên trong ứng dụng. Một hệ thống định tuyến có 2 mục đích:

* Xây dựng một tập hợp các URL đi vào ứng dụng và định tuyến chúng tới các Controller và thực thi các phương thức Action để xử lý.
* Xây dựng các URL gửi đi mà có thể gọi ngược trở lại Controllers/Actions ( ví dụ: form posts, liên kết <a href=“”> và các lời gọi AJAX).

Sử dụng các quy tắc ánh xạ URL để điều khiển URL đi vào và đi ra để tăng tính mềm dẻo cho việc lập trình ứng dụng, nghĩa là nếu muốn thay đổi cấu trúc URL ( ví dụ /Catalog thành /Products ) có thể thay đổi một tập hợp quy tắc ánh xạ mức ứng dụng mà không cần phải viết lại mã lập trình bên trong Controllers và Views

### Các quy tắc định tuyến URL mặc định trong ASP.NET Web Application

Mặc định khi tạo ứng dụng với ASP.NET MVC Web Application trong Visual Studio sẽ tạo ra một ASP.NET Application class gọi là Global.asax chứa cấu hình các quy tắc định tuyến URL. Xây dựng các định tuyến thông qua phương thức RegisterRoutes(ReouteCollection routes) và khi ứng dụng bắt đầu, phương thức Application\_Start() trong Global.asax.cs sẽ gọi RegisterRoutes để tạo ra bảng định tuyến.

Mặc định định tuyến URL trong ASP.NET MVC Framework có cấu trúc dạng: Controllers/ControllerAction/Id .

Với ASP.NET MVC Web Application thì mặc định Controllers là HomeController, mặc định ControllerAction là Index, mặc định Id là rỗng. Nghĩa là khi gọi trang web được xây dựng thông qua template ASP.NET Web Application thì mặc định <http://localhost/> tương đương với <http://localhost/Home/Index/>

Khi ứng dụng ASP.NET MVC Web Application nhận được một Url, MVC Framework sẽ định giá các quy tắc định tuyến trong tập hợp RouteTable.Routes để quyết định Controller nào sẽ điều khiển request.

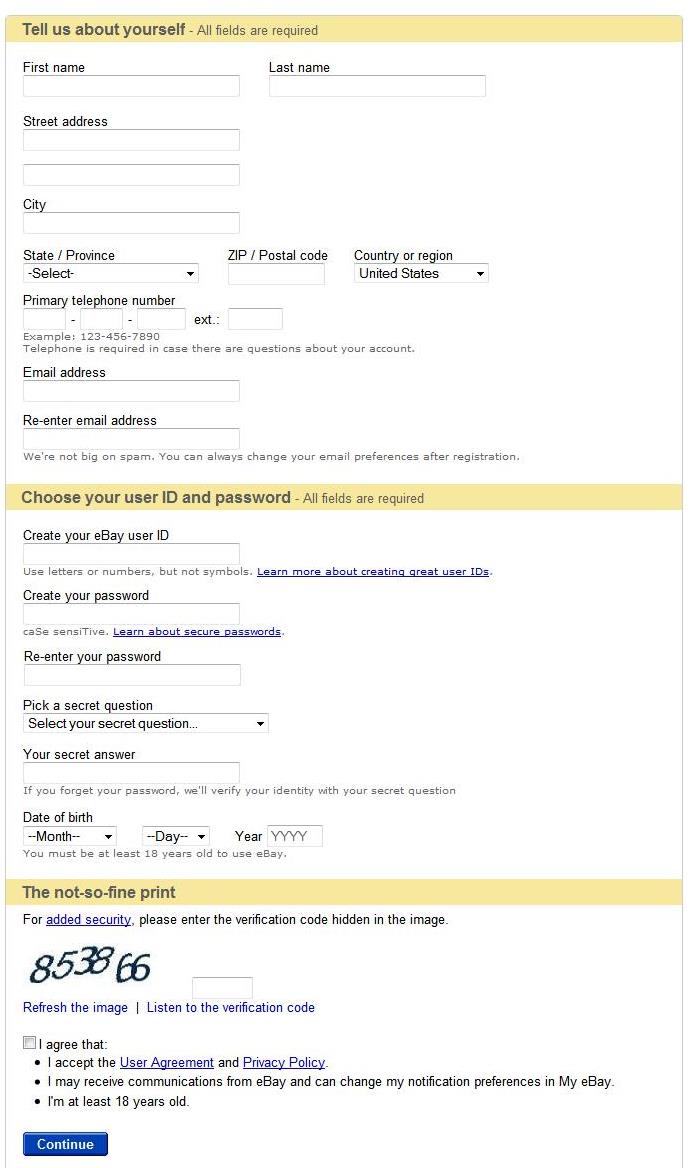
MVC framework chọn Controller bằng cách định giá các quy tắc trong bảng định tuyến theo trật tự đã có sẵn.

# Chương 3. GIỚI THIỆU ĐỀ TÀI

# CHƯƠNG 4. PHÂN TÍCH NGHIỆP VỤ PHÍA NGƯỜI DÙNG

## 4.1 Đăng ký tài khoản (Register)

### 4.1.1 Điền thông tin cá nhân



Hình 3.1 Điền thông tin cá nhân

Để thực hiện giao dịch trên Bidnow người dùng cần đăng ký một tài khoản.

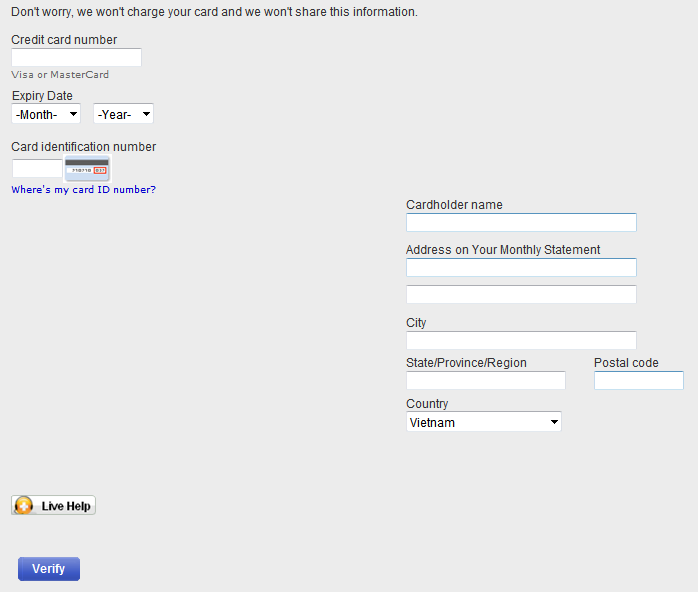
Các việc cần làm ở bước này:

1. Nhập thông tin liên lạc: bao gồm họ tên, địa chỉ nhà, ngày sinh, mã vùng (Postal code, ở USA gọi là Zip Code - mã thư tín - mã bưu điện), tên quốc gia và số điện thoại. Yêu cầu hệ thống kiểm tra xem user đã đủ 18 tuổi hay chưa (dựa vào ngày sinh).
2. Nhập địa chỉ email. Nếu user nhập vào 1 địa chỉ email đã tồn tại trong cơ sở dữ liệu của Bidnow thì hệ thống sẽ thông báo yêu cầu nhập vào 1 địa chỉ email khác.
3. Đăng ký tên đăng nhập (user ID) và mật khẩu.

User sẽ sử dụng user ID và mật khẩu này để đăng nhập vào Bidnow. Hệ thống cũng sẽ thông báo khi user nhập trùng user ID.

1. Đọc và chấp nhận Thỏa thuận thành viên (User Agreement) và Chính sách quyền riêng tư (Private Policy) của BidNow.
2. Điền hình ảnh xác nhận (Verification code). Cho phép user refresh lại hình này.

### 4.1.2 Điền thông tin thẻ tín dụng



Hình 3.2 Điền thông tin thẻ

Như đã đề cập ở chương 3, Bidnow chỉ chấp nhận việc thanh toán tiền mua hàng, tiền thanh toán phí bán hàng(đối với Seller) qua Paypal *(tìm hiểu thêm về Paypal ở phụ lục)*.

Các việc cần làm ở bước này:

1. Điền mã số thẻ. Chỉ chấp nhận thẻ của 2 hãng phổ biến nhất trên thế giới hiện nay là Visa và Master.
2. Ngày hết hạn của thẻ.
3. Mã số nhận dạng thẻ (Card Identification number) gồm 3 chữ số.
4. Tên chủ thẻ
5. Địa chỉ chủ thẻ
6. Mã vùng (Postal code)

Hệ thống sẽ dựa vào thông tin Credit card number để kiểm tra xem tính hợp lệ của thẻ *(Xem thêm ở Phụ lục A)*

Nếu Verify thành công, hệ thống sẽ thông báo user đã hoàn tất quá trình đăng ký và yêu cầu user kích hoạt tài khoản thông qua việc kiểm tra email. Email này do hệ thống tự động gửi đến email mà user đã đăng ký lúc đầu, nó cung cấp một đường dẫn để user kích hoạt tài khoản vừa đăng ký.

Sau khi kích hoạt tài khoản thành công, hệ thống tiếp tục tự động gửi đến user 1 email với nội dung “Welcome to Bidnow!”.

### 4.1.3 Quên userID hoặc mật khẩu

Trong trường hợp user quên userID hoặc mật khẩu, user làm theo quy trình như sau:

1. User nhấn vào liên kết “*forget my password”*, liên kết này chỉ xuất hiện ở mục Đăng nhập (Sign up).
2. Trang web sẽ yêu cầu user nhập:

* Địa chỉ email
* Điền lại hình ảnh xác nhận

1. Trang web sẽ kiểm tra sự tồn tại của emal này trong hệ thống cơ sở dữ liệu và gửi tới email này tên userID tương ứng, kèm theo một đường link (link này tồn tại trong vòng 24 giờ).
2. User theo đường link đó, sẽ được chuyển tới trang yêu cầu nhập

* Password mới
* Hình ảnh xác nhận.

1. Nhấn OK và hoàn tất.

# Phụ Lục A. Kiểm Tra Tính Hợp Lệ Của Một Thẻ Tín Dụng

**1. Tiền tố, Chiều dài và Thuật toán kiểm tra chữ số**

|  |  |  |  |
| --- | --- | --- | --- |
| Loại thẻ | Tiền tố | Chiều dài | Thuật toán |
| MasterCard | 51 - 55 | 16 | Mod 10 |
| VISA | 4 | 16 | Mod 10 |
| AMEX | 34 - 37 | 15 | Mod 10 |

**2. Công thức LUHN (Mod 10)**

Việc xác nhận một thẻ có hợp lệ hay không gồm các bước sau đây:

Bước 1: Nhân đôi giá trị của các số khác nhau của một số tài khoản chính bắt đầu với chữ số thứ hai từ bên phải

Bước 2: Cộng tất cả các chữ số thu được ở bước 1

Bước 3: Kiểm tra nếu tổng số thu được ở bước 2 là một số kết thúc bằng số không (30, 40, 50, …) thì đây là một thẻ hợp lệ

Ví dụ, để xác nhận số tài khoản chính của 49927398716:

Bước 1:

4 9 9 2 7 3 9 8 7 1 6

x2 x2 x2 x2 x2

------------------------------

18 4 6 16 2

Bước 2: 4 + (1 8) + 9 + (4) + 7 + (6) + 9 + (1 6) + 7 + (2) + 6

Bước 3: Tổng = 70: Thẻ này hợp lệ