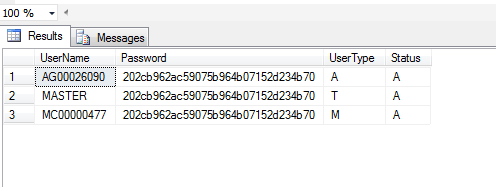
Link down database : <https://www.dropbox.com/s/f6gh92s1371en4s/SERVER_data%20ver%202%204-11-2016.rar?dl=0>  
Source code web: <https://github.com/tudinh29/WebServer>

Chú ý: Các store đã viết sẵn ở dropbox thư mục PM tuần 6. Trên github có 1 file store:chỉ chứa tất cả các store cần thiết cho app web hiện tại . Nên khi code web cần store gì thì vào Pm Tuần 6 lấy rồi copy vô file Store trên github

Cấu hình web config: Vào WebApi 🡪 Web.config kiếm <add name="APIDbContext" rồi chỉnh lại theo dtb máy mình 🡪 Vào WebMVC 🡪 web.config làm tương tự

**Build** : chạy Ctrl+F5 WebAPI 🡪 báo lỗi 403 trước rồi chạy Ctrl +F5 WebMVC.   
hướng dẫn: chuột phải vào project WebAPI chọn Set as startup Project 🡪 bấm Ctrl+F5 🡪 Chuột phải vào project WebMVC chọn Set as Startup Project 🡪bấm Ctrl + F5

**App web :   
Login** : Đăng nhập kiểm tra thông tin trong bảng user\_info. Mọi người tự insert vào để đăng nhập  
lưu ý. Password trong dtb có mã hóa md5. Lên đây để mã hóa nhé : http://www.md5.cz/  
Ví dụ muốn đăng nhập với password là 123 thì gõ 123 vào web trên sẽ ra 202cb962ac59075b964b07152d234b70, copy cái này vào database  
Username phải giống với MerchantCode hoặc AgentCode nhé



**Cơ chế cần biết:**

Trên view click sẽ gọi tới hàm nào của Controller nào của project **WebMVC**: như

@Ajax.ActionLink("ACTIVE", "ChangeStatus", "Management", new { @id = item.AgentCode }, new AjaxOptions

Sẽ gọi tới hàm ChangeStatus trong Controller Management.

Trong hàm ChangeStatus có gọi project WebAPI để kết nối với database:

HttpClient client = new AccessAPI().Access();

StringContent content = new StringContent("");

response = client.PostAsync(string.Format("api/Agent/ChangeStatus?agentCode={0}", id), content).Result;

Ở đấy chuỗi kêt nối là Api/Agent/ChangeStatus? agentCode=.. 🡺 Gọi tới Controller AgentController trong WebAPI, hàm ChangeStatus với tham số là agentCode

[HttpPost]

public bool ChangeStatus(string agentCode)

{

try

{

object[] paremeter =

{

new SqlParameter("@AgentCode", agentCode)

};

db.Database.ExecuteSqlCommand("exec sp\_InactiveOrActive\_Agent @AgentCode", paremeter);

return true;

}

catch(Exception)

{

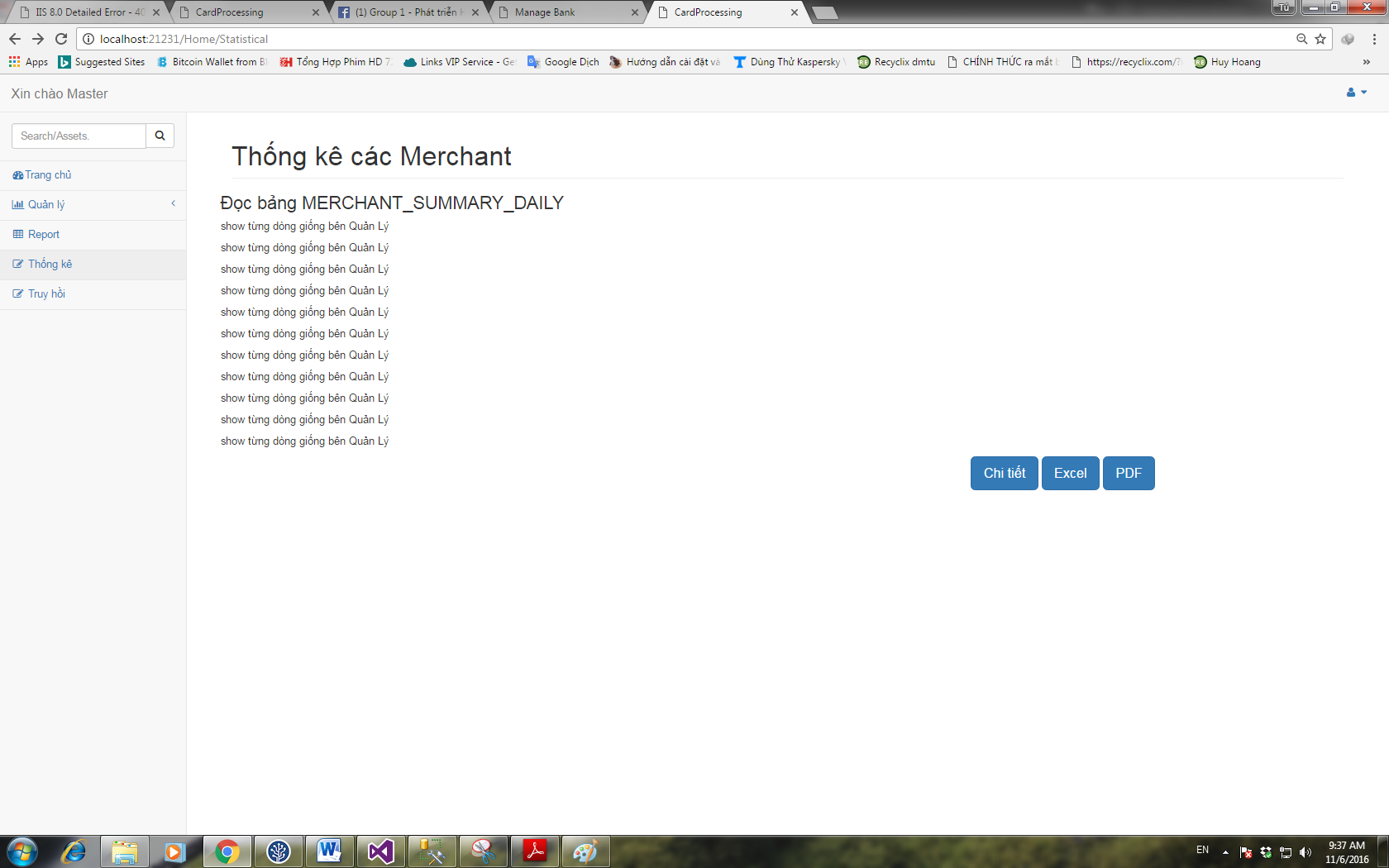
return false;

}

}

Tương tự các hàm FindAgent là return res chứa các agent. Khi gọi dùng GetAsync để lấy thay vì PostAsync là update

CHỨC NĂNG

1. Thống kê   
   

Tham khảo code bên phần quản lý : ManagementController

Đọc dữ liệu bảng Summary\_Daily 🡪 load lên view. Các cột cần load [ReportDate]

,[MerchantCode]

,[SaleAmount]

,[SaleCount]

,[ReturnAmount]

,[ReturnCount]

,[NetAmount]  
,[TransactionCount]

,[KeyedCount]

Code trong **HomeController**, ở public ActionResult Statistical() c phân quyền 2 người dùng.   
nếu là Master: Type=’T’ thì hiện hết bảng summary. Nếu là Agent: Type =’A’ thì thêm vào where agentCode = temp.UserName. K CÓ MERCHANT

MERCHANT

1. Button chi tiết

Trỏ tới action Detail của Controller Statistical

@Html.ActionLink("Chi tiết", "Detail", "Statistical", new { listMerchant = Model }, new { @class= "btn btn-primary btn-lg"})

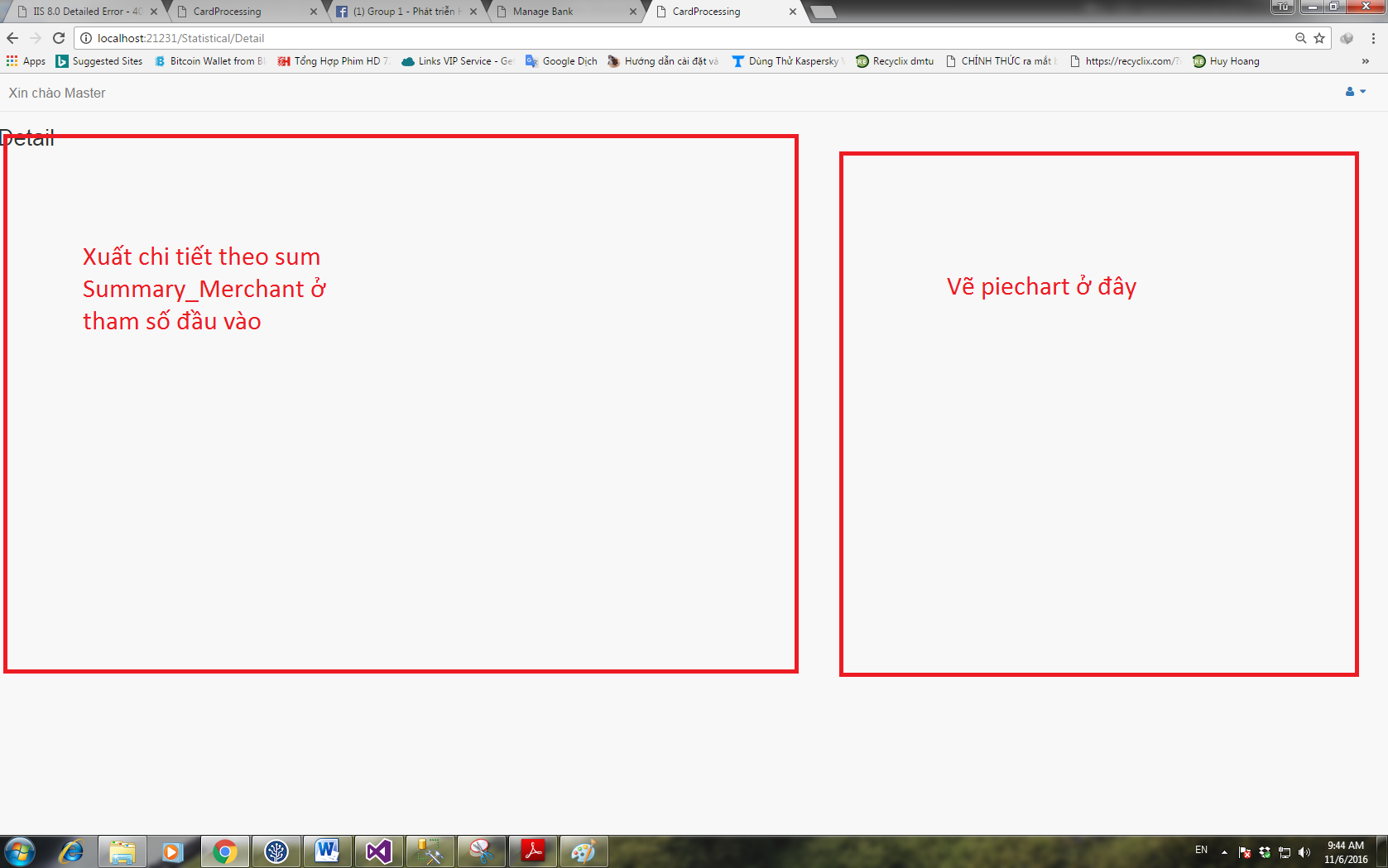
public ActionResult Detail(List<MERCHANT> listMerchant)

{

return View();

}

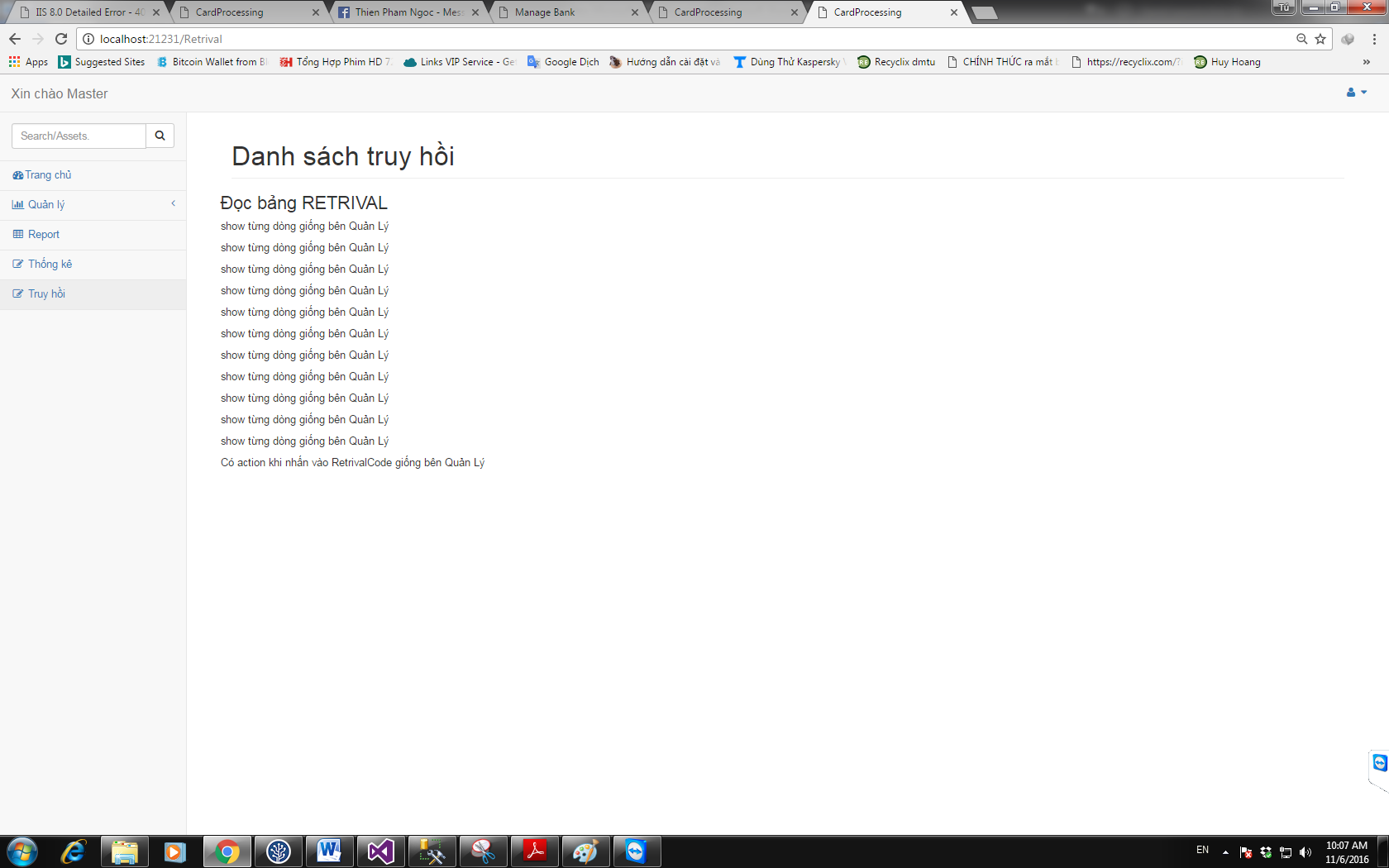
T Tham số đầu vào là List Summary\_Merchant . Sum lại thành 1 dòng( rồi xuất ra tất cả các côt)



1. Button PDF và Excel

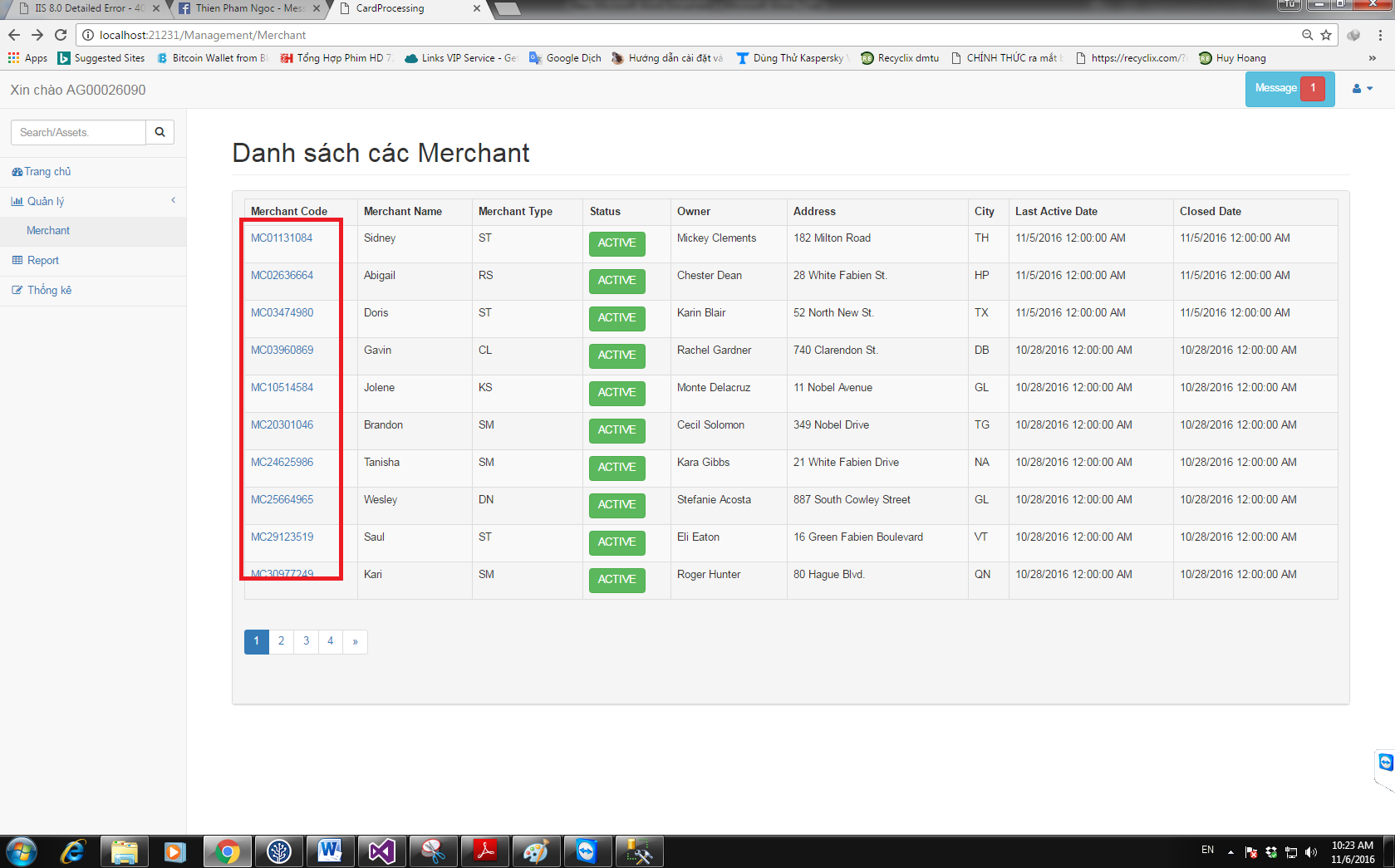
Dựa vào data trên view xuất ra. Tham khao https://www.youtube.com/watch?v=jXP3achaGbA

1. Truy hồi



1. Message

Nhóm thiện làm

1. Click view vào actionLink  
   

Tham khảo View profile trong Merchant/Agent/Edit. Làm y chang.

Hiện đầy đủ thông tin và thêm chức năng Edit + save cập nhật thông tin

1. Chức năng tìm kiếm trong controller Quản lý

Tham khảo :

https://www.youtube.com/watch?v=IfiC1Ye9qNc&list=PLRhlTlpDUWsyK1TIsewrQ7WwC7QkCSCPD&index=21