**Gamestop.com Development Environment Setup**

1. **Equipments – test copy**
2. **Set up Laptop with Wireless (& Wire) Connection** – Manager & Support Team

* Use “GS-CORP” (Gamestop Corporate Network) Gamestop office
* Use “SWGUEST” in Systemware office (Systemware Admin)

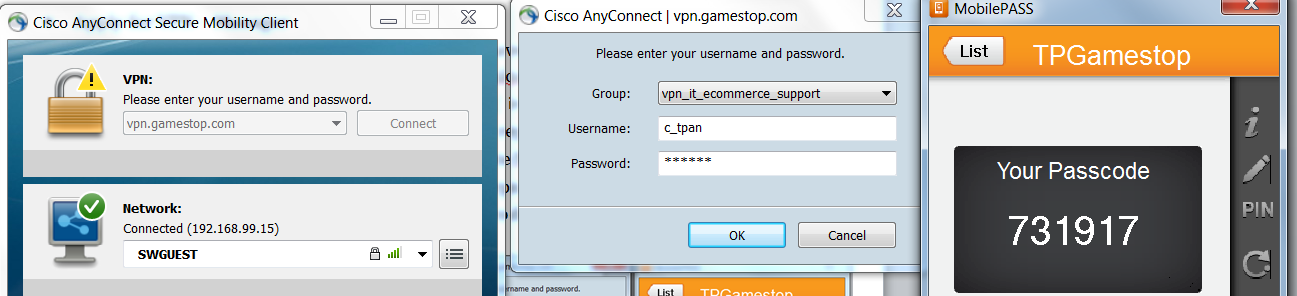
1. **Set up Account babgsetc\xxx** – Manager & Support Team
2. **Configure Outlook and OWA**

* Use the above domain account
* OWA -https://owa.gamestop.com/owa/auth/logon.aspx?replaceCurrent=1&url=https%3a%2f%2fowa.gamestop.com%2fecp%2f%3frfr%3dowa%26p%3dRulesEditor%2fInboxRules.slab

1. **Set up Lync**

* If Lync does not work, email Corporate Help Desk

1. **Set up VPN**

* Apply for VPN – Manager
* Cisco AnyConnect Secure Mobility Client should have been installed
* Corporate Help Desk will send a MobilePass downloand link <http://www.safenet-inc.com/support-downloads/mobilepass-download-page/> in email and the guide for install
* Follow the guide and send the Activation Code by replying the previous email
* After receiving email from Corp Help Desk, activate MobilePass
* Start VPN by running “Cisco AnyConnect” , selecting “vpn.gamestop.com” and then ”vpn\_it\_ecommerce\_support” and entering passcode in the popup generated from MobilePass
* 
* You should be connected to Gamestop VPN
* VPN client is located in C:\Program Files (x86)\Cisco\Cisco AnyConnect Secure Mobility Client\vpnui.exe

1. **Get Rally Access**

* Add email to Rally corporate account - Manager
* https://rally1.rallydev.com/slm/login.op

1. **Get Clarity (Project and Time Reporting)**

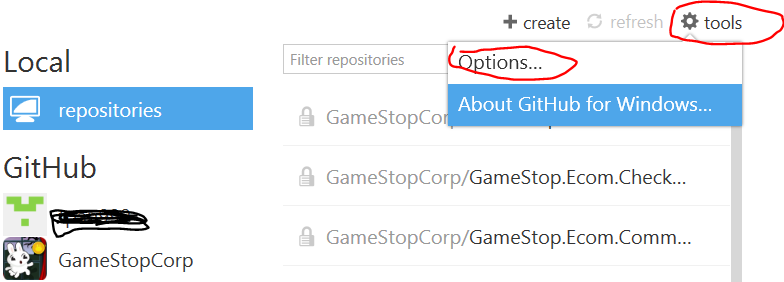
* Add id into the system – Manager
* https://clarity.babgsetc.pvt/niku/nu#action:homeActionId

1. **Get TeamCity (Build)**

* Add id into the sytem – Manager
* <http://cibuilds.gamestop.com:8080/profile.html>

1. **Set up Lync** - Corporate Help Desk (email)
2. **Environment**

* **Set up GitHub**
* Create a Github account from http://www.github.com
* Add it to Gamestop Corporate Account “GameStopCorp” - Manager
* Download and install GitHub client from <https://windows.github.com>
* Launch “**All Programs->GitHub, Inc->GitHub**” and login to the client using the above account
* From “**Tools->Options**”, set “**default storage directory**” to C:\GameStop\Source\ (or any folder)



* Find “GamestopCorp/Tools.SetupScript” Repository or any ones and click on “Clone”
* Optional: use the script to clone source
  + - launch Git Bash CMD
    - run C:\Gamestop\Source\Tools.SetupScripts\git-setup.sh to get all required source files
* Download Git Bash <http://git-scm.com/download/win/>
* From All Programs/Git, click on Git Bash to get into Git Bash CMD
* GitHub beginner site: <http://readwrite.com/2013/10/02/github-for-beginners-part-2#awesm=~oCaAbLBXEd7fvE>
* Another useful tool TortoiseGit at <https://code.google.com/p/tortoisegit/>
* **Turn on IIS in Windows 7**
* Go to “Control Panel / Program / Turn Windows Features On or Off”
* Turn the following Windows Features of “Internet Information services” on:
  + - World Wide Web Services (all options)
    - Web Management (all options)
* If using ASP.Net 4.0, you might need to register ASP.Net 4.0
  + - Run CMD as Administrator and then execute
    - C:\Windows\Microsoft.NET\Framework\v4.0.30319\aspnet\_regiis -i
* **Set up Commerce Server 2007**
* The account (for example babgsetc\xxx) being used needs to be added into the Commerce Server Administration SQL server user group - Manager
* Install Commerce Server 2007 Developer Edition

<http://www.microsoft.com/en-us/download/details.aspx?id=17650> and keep default selections (greyed out ones)

* Install Commerce Server 2007 Service Pack 2

<http://www.microsoft.com/en-us/download/details.aspx?id=9046>

* Install HotFix KB975859

<http://support.microsoft.com/kb/975859>

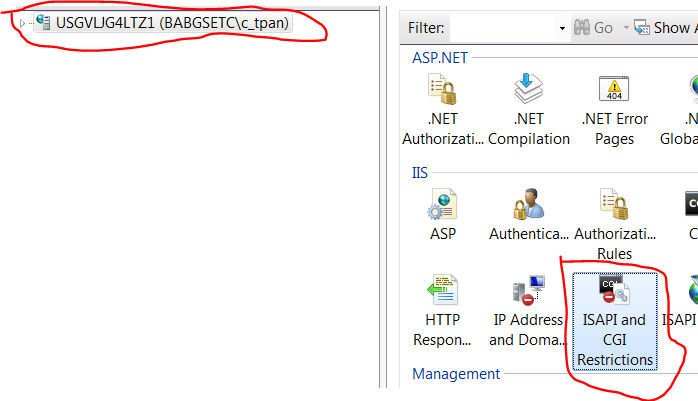
* Open Commerce Server Configuration Wizard (**All Programs->Microsoft Commerce Server 2007->Tools**) and specify
  + - Administrative Server: dl1gsddb01.testecom.pvt\commerce
    - Window Authentication: the above account ID
    - Optional: in Configuration Wizard, load the file CSConfig.xml with the following:

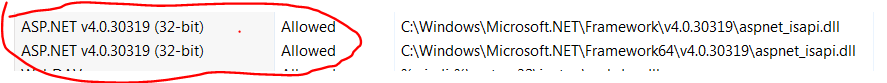
<Configuration>  
< SQL ID="CommerceAdminDB">  
< Server>dl1gsddb01.testecom.pvt\commerce</Server>  
< Database>MSCS\_Admin</Database>  
< WindowsSecurity>yes</WindowsSecurity>  
< Password />  
< /SQL>  
< /Configuration>

* Run the following to register Commerce Server Profile Keys
  + - C:\Program Files (x86)\Microsoft Commerce Server 2007\Tools\ProfileKeyManager.exe /kn /f /reg "HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Commerce Server 2007 Keys\GameStop"
    - C:\Program Files (x86)\Microsoft Commerce Server 2007\Tools\ProfileKeyManager.exe /kn /f /reg "HKEY\_LOCAL\_MACHINE\SOFTWARE\Wow6432Node\Microsoft\Commerce Server 2007 Keys\GameStop"
* **Install Visual Studio 2012 Professional**
* Get package from Manager
* Use Pismo (or others) to mount ISO file.
* **Set Host File for Web Applications**
* In C:\Windows\System32\drivers\etc, add these Entries  
  127.0.0.1 local.gamestop.com  
  127.0.0.1 local.m.gamestop.com  
  127.0.0.1 local.services.gamestop.com  
  127.0.0.1 local.pur.gamestop.com  
  127.0.0.1 local.purm.gamestop.com  
  127.0.0.1 dl1gsdweb60.testecom.pvt
  + - 1. local.login.gamestop.com

1. **Set up Web Sites**
2. **Turn on ISAPI**

* In IIS Management Console, click on the machine name and then click on icon “ISAPI and CGI Restriction” and make ASP.NET v4.xxx “Allowed”.





1. **Create SSL Certificates**

* In IIS Management Console, double click on the machine name and then double click on “Server Certificates” in the middle panel. On the right panel, click on “Create Self-Signed Certificate…”.
* Specify the certificate name, for example “local.www.gamestop.com” and then “OK”.
* Create several this kind of certificates for future use:

local.www.gamestop.com

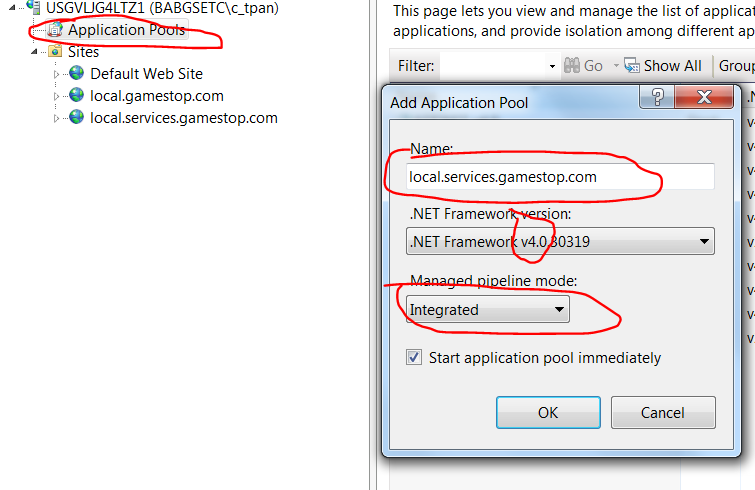
local.m.gamestop.com

local.services.gamestop.com

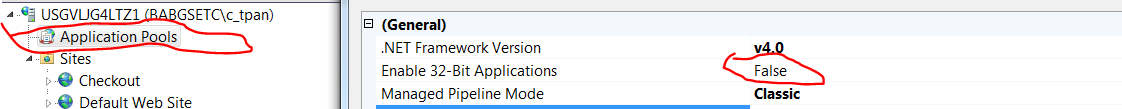
local.login.gamestop.com

1. **Create Application Pools**

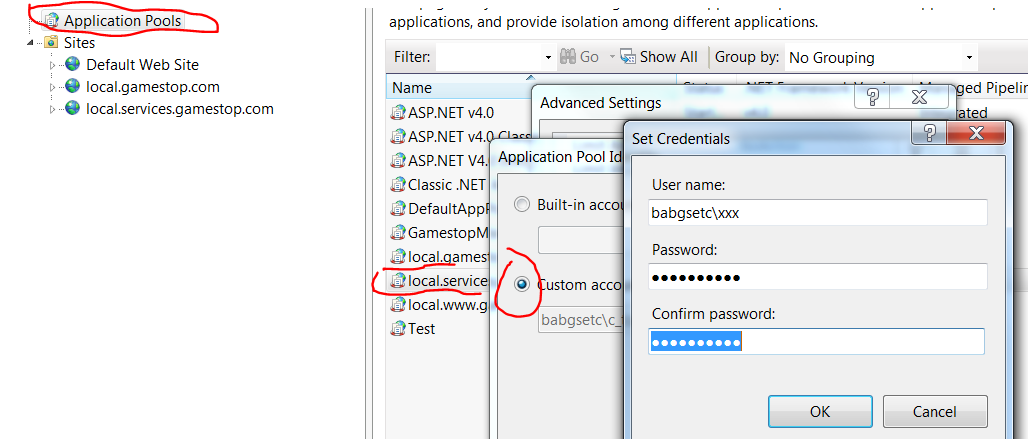
* In IIS Management Console, click on “Application Pool” and then right click on it to select “Add Application Pool”. Add a new pool “local.services.gamestop.com” with framework v4.xxx and pipeline mode “Integrated”.



* It this pool’s Advanced Setting, set “Enable 32 Bit Applications” to “False”



* In the same screen, select Identity to be “customer account” and use “babgsetc\xxx”. Then click on “OK”.



* Create several application pools using the above steps for future use:

local.services.gamestop.com -> for site local.services.gamestop.com

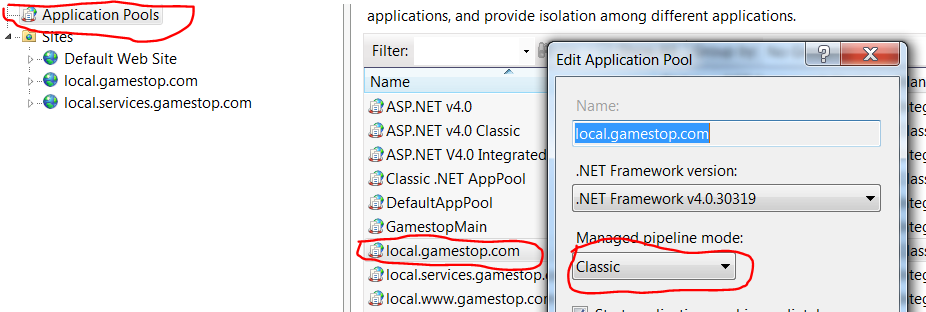
local.m.gamestop.com -> for site local.m.gamestop.com

local.login.gamestop.com -> for site local.login.gamestop.com

local.www.gamestop.com -> for site local.gamestop.com/checkout

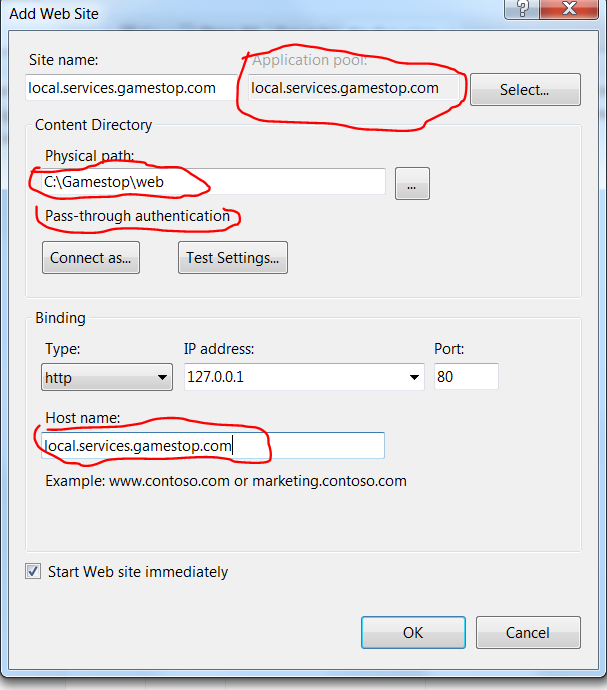
local.gamestop.com -> for site local.gamestop.com

**Note:** for local.gamestop.com application pool only, use “Classic” pipeline mode:

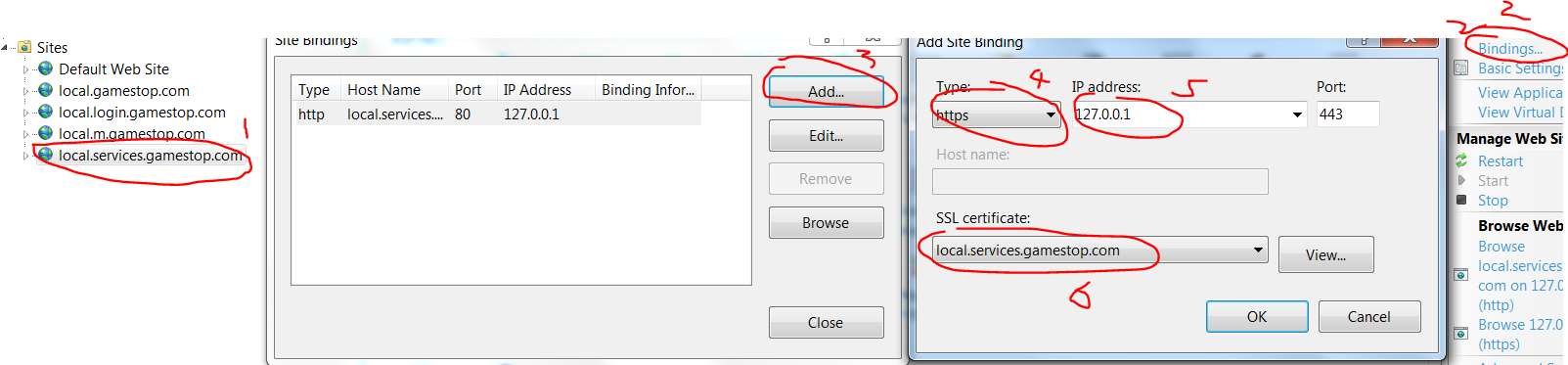


1. **Create Services Site**

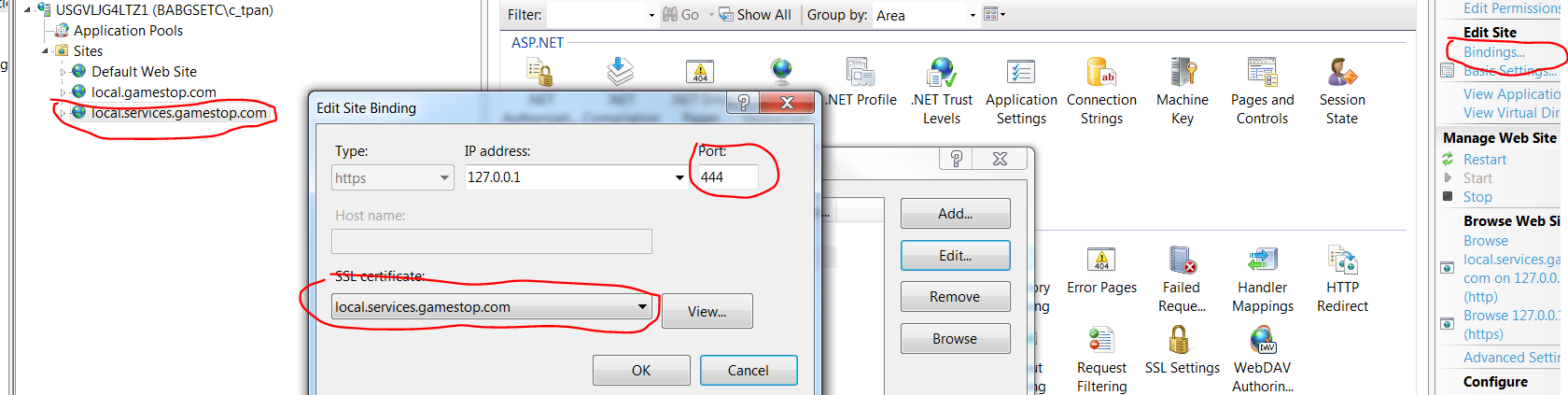
* Create an empty folder “web” under C:\Gamestop.
* In IIS Management Console, right click on “**Sites**” and “**Add Web Site**” “local.services.gamestop.com” with physical path to “C:\Gamestop\web\”
* Select “local.services.gamestop.com” from the Application pool dropdown and keep “Pass-through authentication”
* Use “local.services.gamestop.com” as the host name with IP 127.0.0.1



* Double click on site “local.services.gamestop.com” and click on Binding on the right panel. Add https with SSL certificate “local.services.gamestop.com” and local IP “127.0.0.1” and port “443”.



* + - this is especially for Customer/Account Service which requests <https://local.services.gamestop.com/Ecom/Customers/v1/AccountService.svc>. In local, ignore security warning with this SSL setting.
    - Note: due to the local environment, port “443” for this site might need to be changed to give it to another site port “443” is needed for test. See Section “Tricks” for detail.
    - Another way: use the same IP and SSL certificate with different port to avoid conflicting with local.gamestop.com which uses the same IP and SSL certificate with port 443. However, this might need the network team to grant port access.



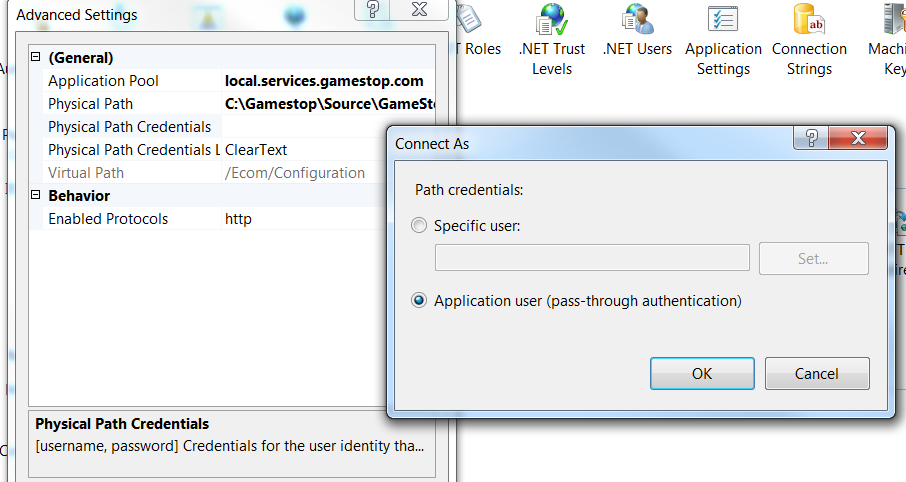
* Under “C:\Gamestop\web\, create a folder “Ecom” which will hosts service applications

1. **Build Service Applications**
   * Services solutions are in the following folders. To test services, they have to be rebuilt.

|  |  |  |
| --- | --- | --- |
| **Service (Application in IIS)** | **Physical Path C:\Gamestop\Source\** | **Service Solution Folder**  **C:\Gamestop\Source\** |
| Configuration | GameStop.Ecom.Configuration\GameStop.Ecom.Configuration.Data | GameStop.Ecom.Configuration\Gamestop.Ecom.Configuration.sln |
| Configuration  /Data | GameStop.Ecom.Configuration\GameStop.Ecom.Configuration.Data | One project with the above solution |
| Customers | GameStop.Ecom.Customers\GameStop.Ecom.Customers.Account | GameStop.Ecom.Configuration\Gamestop.Ecom.Customers.sln |
| DigitalWallet | GameStop.Ecom.DigitalWallet\GameStop.Csl.DigitalWallet | GameStop.Ecom.DigitalWallet\GameStop.Csl.DigitalWallet.sln |
| Fulfillment | GameStop.Ecom.Orders\GameStop.Ecom.Orders.Fulfillment.Digital | GameStop.Ecom.Orders\GameStop.Ecom.Orders.sln |
| Merchandising | GameStop.Ecom.Merchandising\GameStop.Ecom.Merchandising | GameStop.Ecom.Merchandising\GameStop.Ecom.Merchandising.sln |
| Orders | GameStop.Ecom.Orders\GameStop.Ecom.Orders | GameStop.Ecom.Orders\GameStop.Ecom.Orders.sln |
| PaymentAndFraud | C:\Gamestop\Ecom\ | Virtual directory – or point to an empty folder |
| PaymentAndFraud  /Payment | GameStop.Ecom.PaymentAndFraud\GameStop.Ecom.PaymentAndFraud.Payment | GameStop.Ecom.PaymentAndFraud\GameStop.Ecom.PaymentAndFraud.sln |
| Stores | GameStop.Ecom.Stores\GameStop.Ecom.Stores.Search | GameStop.Ecom.Stores\GameStop.Ecom.Stores.sln |
| Tax | GameStop.Ecom.Orders\GameStop.Ecom.Orders.Tax | GameStop.Ecom.Orders\GameStop.Ecom.Orders.sln |
| WebOrder | GameStop.Ecom.Orders\GameStop.Ecom.Orders.WebOrder | GameStop.Ecom.Orders\GameStop.Ecom.Orders.sln |

* Launch Visual Studio 2012 with Administrator
* For example, open Gamestop.Ecom.Configuration.sln and build with success

1. **Add Services to Service Site**

* In IIS Management Console, select web site “local.services.gamestop.com”. Right click on “Ecom” -> Add Application “Configuration” and physically point to C:\Gamestop\source\GameStop.Ecom.Configuration\GameStop.Ecom.Configuration.Data
* In Basic Setting, select Application pool “local.services.gamestop.com” and “pass-through authentication”
* In Advanced Setting, select Physical Path Credentials as below:
* Testing:
  + - Select local.services.gamestop.com/Ecom/Configuration
    - On the bottom of the middle panel, click on Content View and the click on folder “v1” to see service list.
    - Browse .svc file to see the WCF service definition.
  + Under “Configuration” site, add a new application “Data” and point to GameStop.Ecom.Configuration\GameStop.Ecom.Configuration.Data and set up the same as Configuraiton.
  + Repeat this for all other services. Assume:
    - all source code paths are under C:\Gamestop\Source\ and
    - all services are under http://local.services.gamestop.com/Ecom/

|  |  |  |
| --- | --- | --- |
| **Service (Application in IIS)** | **Physical Path C:\Gamestop\Source\** | **Service to Test** [**http://local.services.gamestop.com/Ecom/**](http://local.services.gamestop.com/Ecom/) |
| Configuration | GameStop.Ecom.Configuration\GameStop.Ecom.Configuration.Data | /Configuration/v1/ConfigurationDataService.svc |
| Configuration  /Data | GameStop.Ecom.Configuration\GameStop.Ecom.Configuration.Data | /Configuration/Data/v1/ConfigurationDataService.svc |
| Customers | GameStop.Ecom.Customers\GameStop.Ecom.Customers.Account | <https://local.services.gamestop.com/Ecom/Customers/v1/AccountService.svc>  http://local.services.gamestop.com/Ecom /Customers/v1/ProfileService.svc |
| DigitalWallet | GameStop.Ecom.DigitalWallet\GameStop.Csl.DigitalWallet | /DigitalWallet/v1/DigitalWalletService.svc |
| Fulfillment | GameStop.Ecom.Orders\GameStop.Ecom.Orders.Fulfillment.Digital | /Fulfillment/v1/DigitalContentService.svc |
| Merchandising | GameStop.Ecom.Merchandising\GameStop.Ecom.Merchandising | /Merchandising/v1/BarcodeService.svc |
| Orders | GameStop.Ecom.Orders\GameStop.Ecom.Orders | /Orders/v1/CartService.svc |
| PaymentAndFraud | C:\Gamestop\Ecom\ | Virtual directory – or point to an empty folder |
| PaymentAndFraud  /Payment | GameStop.Ecom.PaymentAndFraud\GameStop.Ecom.PaymentAndFraud.Payment | /Ecom/PaymentAndFraud/Payment/v1/IntegrationPaymentService.svc |
| Stores | GameStop.Ecom.Stores\GameStop.Ecom.Stores.Search | /Store/v1/StoreSearchService.svc |
| Tax | GameStop.Ecom.Orders\GameStop.Ecom.Orders.Tax | /Tax/v1/TaxService.svc |
| WebOrder | GameStop.Ecom.Orders\GameStop.Ecom.Orders.WebOrder | /WebOrder/v1/WebOrderService.svc |

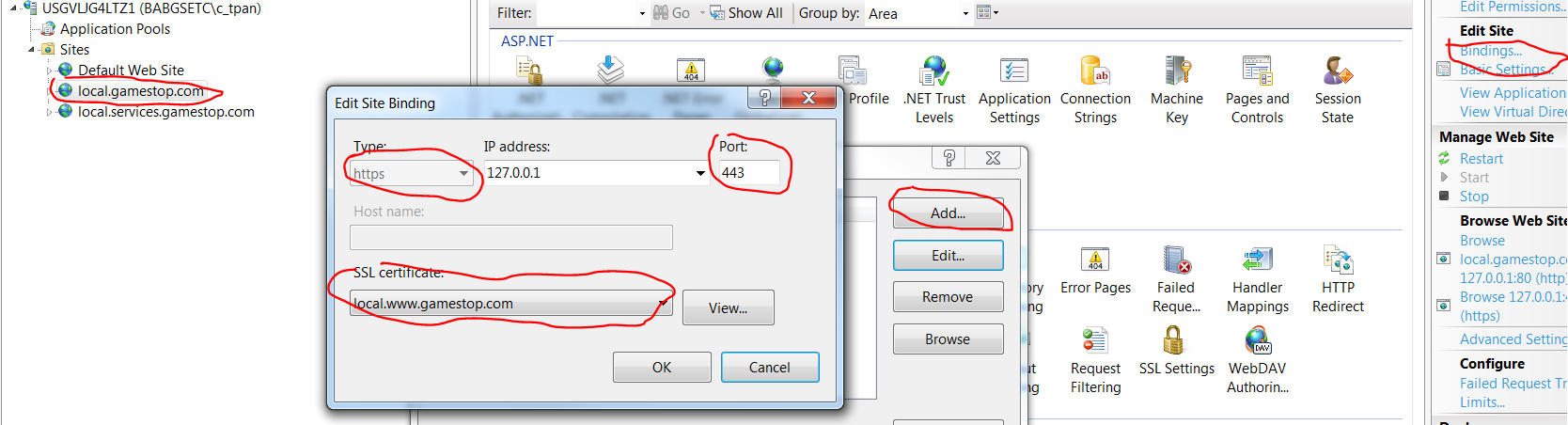
1. **Create local gamestop.com**

* Open following solutions to build to success.

**Note:** /searchmerch/ site depends on “local.gamestop.com” for build. So, this site might need to be created as below before successfully building it.

|  |  |  |
| --- | --- | --- |
| **Site** | **Physical Path** | **Solution Folder**  **C:\Gamestop\Source\** |
| local.gamestop.com | C:\Gamestop\Source\Gamestop.com\GameStop\MainSite | Gamestop.com\GameStop\Gamestop.sln |
| /checkout | C:\Gamestop\Source\GameStop.Ecom.Checkout\GameStop.Ecom.Checkout.UI | GameStop.Ecom.Checkout\Gamestop.Ecom.Checkout.sln |
| /searchmerch | C:\Gamestop\Source\Gamestop.mvc\Gamestop | C:\Gamestop\Source\Gamestop.mvc\Gamestop.sln |
| /gs | \\dl1gsdweb01.testecom.pvt\Site Resources\gs | Virtual Directory Only |

* In IIS Management Console, right click on “**Sites**” and “**Add Web Site**” “local.gamestop.com” with physical path to “C:\Gamestop\Source\Gamestop.com\GameStop\MainSite”
* Select “local.gamestop.com” from the Application pool dropdown and keep “Pass-through authentication”
* Use “local.gamestop.com” as the host name with IP 127.0.0.1 and “http”
* Double click on site “local.gamestop.com” and click on Binding on the right panel. Add https with SSL certificate “local.gamestop.com” and local IP “127.0.0.1” and port “443”. This is for sub site /checkout/ which needs https.



* Add sub application sites /checkout/, /searchmerch/ using application pool “local.www.gamestop.com”, and virtual directory site /gs/ under “local.gamestop.com” below:

|  |  |  |
| --- | --- | --- |
| **Site** | **Physical Path** | **URL to Test** |
| local.gamestop.com | C:\Gamestop\Source\Gamestop.com\GameStop\MainSite | http://local.gamestop.com/Default.aspx |
| /checkout | C:\Gamestop\Source\GameStop.Ecom.Checkout\GameStop.Ecom.Checkout.UI | https://local.gamestop.com/checkout |
| /searchmerch | C:\Gamestop\Source\Gamestop.mvc\Gamestop | http://local.gamestop.com/searchmerch/header |
| /gs | \\dl1gsdweb01.testecom.pvt\Site Resources\gs | Virtual Directory Only |

1. **Create local.m.gamestop.com**

* Create an application site “local.m.gamestop.com” using the application pool “local.m.gamestop.com” with the following physical path. And also create sub site /checkout/

|  |  |  |
| --- | --- | --- |
| **Site** | **Physical Path** | **URL to Test** |
| local.m.gamestop.com | C:\Gamestop\Source\Gamestop.com\GameStop\GameStop.Mobile | http://local.m.gamestop.com/ |
| /checkout | C:\Gamestop\Source\GameStop.Ecom.Checkout\GameStop.Ecom.Checkout.UI | https://local.m.gamestop.com/checkout |

* “local.m.gamestop.com” is part of solution C:\Gamestop\Source\ Gamestop.com\GameStop\Gamestop.sln.
* “local.m.gamestop.com/checkout/” points to the same location as “local.gamestop.com/checkout/” and in the same solution.
* Bind SSL using “local.m.gamestop.com” certificate to “local.m.gamestop.com” which needs https for “Order Lookup” and “Cart”.

1. **Create local.login.gamestop.com**

* Create an application site “local.login.gamestop.com” using the application pool “local.login.gamestop.com” with the following physical path.

|  |  |  |
| --- | --- | --- |
| **Site** | **Physical Path** | **URL to Test** |
| local.login.gamestop.com | C:\Gamestop\Source\GameStop.Profile\GameStop.MultiPass.IssuingProvider.Web | https://local.login.gamestop.com/Account/Login |

* The solution is located at C:\Gamestop\Source\GameStop.Profile\Gamestop.MultiPass.IssuingProvider.sln.
* To successfully build the solution, perform the following things.
* Run Visual Studio as Administrator
  + - In Visual Studio, enable “NugetPackageRestore” by checking the box under: Tools > Options > Package Manager > Allow NuGet, which will download missing packages during build
* Download and install “NodeJS” from <http://nodejs.org>.
  + - Got to C:\Program Files\nodejs and run

npm install transcend –g

* + - This is to install “Transcend”. Transcend is a framework for including and minimizing JavaScript for a single page app written in NodeJS by Alex Brombal.
    - Once installed, transcend will run automatically every time the project builds. If you want to have Transcend monitor the javascript files and run every time a change is made, without having to do a build, you can navigate to GameStop.MultiPass.IssuingProvider.Web/js and run the following command:

transcendjs --watch --config=transcend.debug.json ./transcend ./build

* + - Transcend reads from the /transcend folder and writes to the /build folder. You should NEVER modify files in the the /build folder manually because your changes will be overritten the next time transcend runs.

#### Make SCSS/Compass work

* + - Download Ruby from <http://www.ruby-lang.org/en/downloads/>
    - Use 7-zip to unzip the downloaded Ruby file.
    - Right click “Computer” and select “Property->Advanced system settings->Environment Variables”. Select “PATH” variable to “Edit”. Add Ruby bin path to “PATH” variable. For example, the bin folder is C:\Gamestop\Software\ruby-1.9.3-p545-i386-mingw32\ruby-1.9.3-p545-i386-mingw32\bin.
    - Ruby will make use “SCSS and Compass” from the command line
    - Install “Compass gem”

C:\any\_directory> gem install compass

* + - Navigate to the GameStop.MultiPass.IssuingProvider.Web directory, and run the following command to watch the directory for changes:

compass watch -s nested

* + - run a one-time compile, you can use “compile” instead of “watch”
    - SCSS is the CSS preprocessor we use to make CSS development easier. Compass is a framework that includes many cross-browser compatibilities and shortcuts.
    - You don't need to install SCSS separately; it is included with Compass.

1. **Git Hub and Team City**

* Guide for source code management using Git Hub is in another document.
* Guide for push and build process is in another document.

1. **Tricks**

* Use QA
  + - local.gamestop.com/checkout uses AccountService. But both of them needs https:443. However, only one can have simultaneously. Gamestop.Ecom.Checkout.UI web.conf, use https and SSL binding configuration
    - Solution: in Checkout web.config, use QA AccountService as the endpoint as below:

<endpoint address="**https**://**qa**.services.gamestop.com:**443**/Ecom/Customers/v1/AccountService.svc" binding="wsHttpBinding" bindingConfiguration="WSHttpBinding\_Default**SSL**" contract="AccountService.AccountService" name="WSHttpBinding\_AccountService" />

* Switch Port 443
  + - Because in the local environment, only one web site can have port 443. So, you have to change port 443 to another number if another site uses it and you might want to stop the other site. If the end-to-end testing needs multiple sites with https, you might do as the above. Hopefully, this issue can be resolved soon by allowing multiple IP addresses for a machine.
* Checkout Billing Info
  + - To make the checkout complete in local.gamestop.com:
    - Billing & Shipping Address: use “**Accept**” as first name. (other options – “Reject” – order be rejected, “Review” – order be reviewed by Fraud)
    - Credit Card

Number: 4111 1111 1111 1111

Expiration: 01/2017 (future month and year)

Security Code: 111