Web Testing

Functional Testing for the NEXTSTEP’s new website and examine the new website for the Top 5 vulnerabilities outlined in the “OWASP Top 10 Risks”

Functional and Security Testing

Table of Contents

[WEB TESTING 2](#_Toc389891702)

[Functional Testing 2](#_Toc389891703)

[Functional Specification 2](#_Toc389891704)

[Testing Steps 2](#_Toc389891705)

[Navigation - Navigate major parts of NEXTSTEP website 2](#_Toc389891706)

[Contact Form 4](#_Toc389891707)

[Shopping Cart - View shopping cart, edit cart content and checkout 5](#_Toc389891708)

[Insert Category 5](#_Toc389891709)

[Edit Category 6](#_Toc389891710)

[Delete Category 6](#_Toc389891711)

[Insert Item 7](#_Toc389891712)

[Edit Item 7](#_Toc389891713)

[Delete Item 8](#_Toc389891714)

[Find Products - find all products 8](#_Toc389891715)

[Find Products - by category 9](#_Toc389891716)

[Find Products - with product name and category 9](#_Toc389891717)

[Login - Fail login 9](#_Toc389891718)

[Login - Successful login 10](#_Toc389891719)

[Security Testing 10](#_Toc389891720)

[A1 – Injection 10](#_Toc389891721)

[Using string concatenation is dangerous for SQL command preparation. Use parameter instead, that will help you get rid of the SQL injection mare. 12](#_Toc389891722)

[A2 – Broken Authentication and Session Management 12](#_Toc389891723)

[A3 – Cross-Site Scripting (XSS) 13](#_Toc389891724)

[A4 – Insecure Direct Object References 14](#_Toc389891725)

[A5 – Security Misconfiguration 17](#_Toc389891726)

[Other ASP.NET security issues 20](#_Toc389891727)

[appendix 23](#_Toc389891728)

[Selenium Scripts for this testing 23](#_Toc389891729)

# WEB TESTING

## Functional Testing

This test is a black-box test that based on the specification of the project’s PID (Project Initiation Documentation). Functions are tested by feeding them input and examining the output, and internal program structure is not considered.

Functional Specification

* Items can be added to a shopping cart for purchase
* The customer may purchase as many items as they like
* The customer may search for an item by selecting the category
* The customer may search for an item by providing the item name or part of the name
* Paying with credit card
* Staff members can login using their username and password
* Staff members will have the ability to add and modify categories
* Staff members will have the ability to add and modify items
* Staff member should have the ability to update their own password
* The staff member must be able to choose their own theme
* Include some jQuery in this site to enhance the user experience.
* Website needs to function well in IE9+, FF and Chrome
* Website needs to validate against W3C HTML 5 and CSS
* Customer can contact with us via this site
* Provide a promotion video on home page

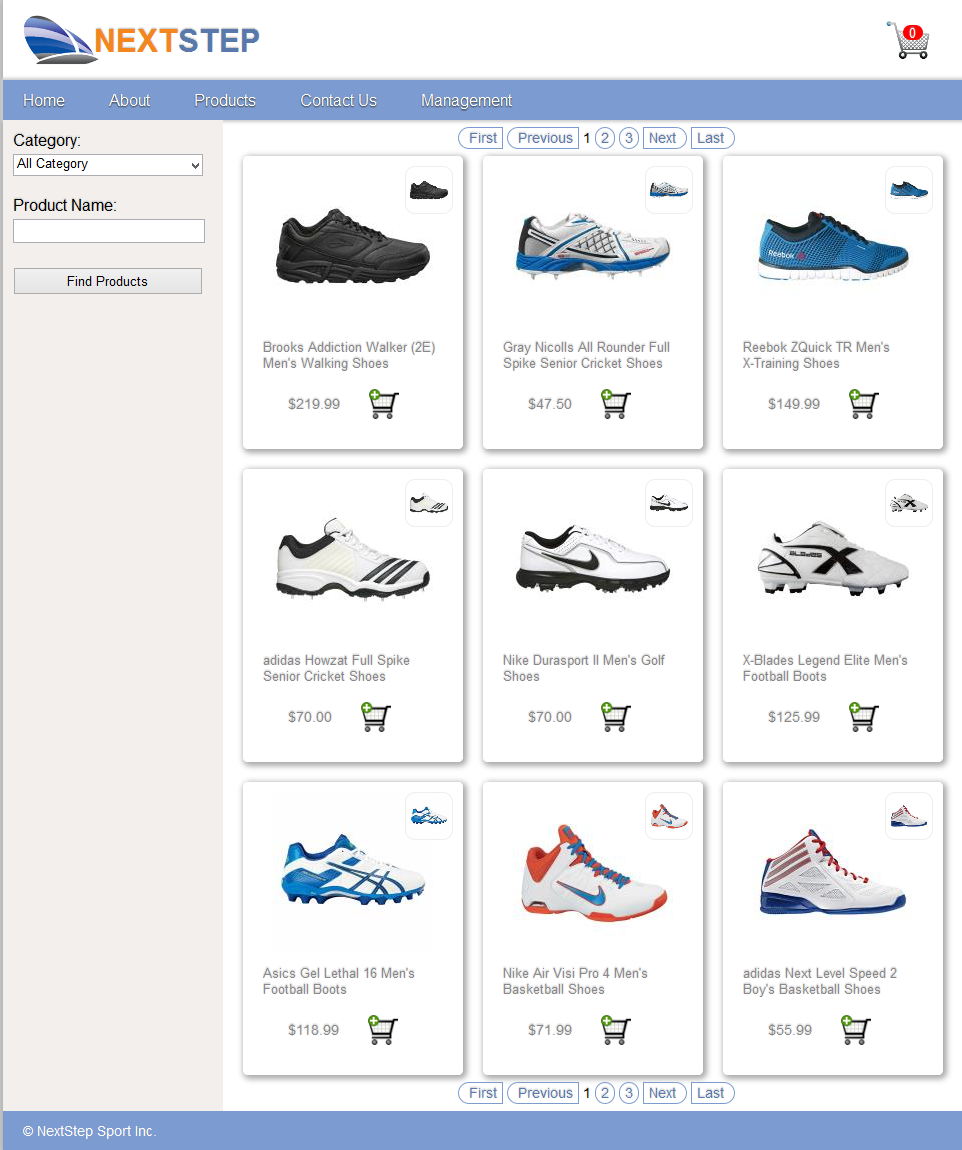
Testing Steps

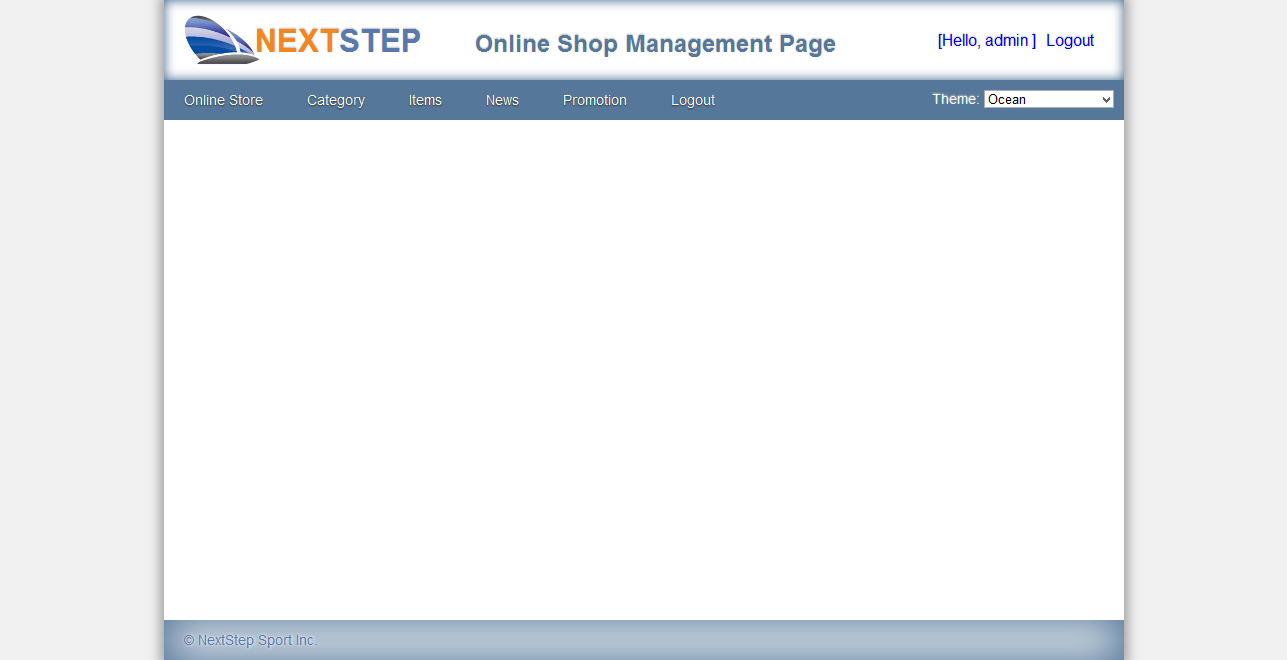
In the functional testing, we use Selenium for the testing tool. Selenium is a suite of tools to automate web browsers across many platforms which can be used in black box website testing. The functional testing will be divided into 14 parts. All scripts are included in the appendix.

Navigation - Navigate major parts of NEXTSTEP website

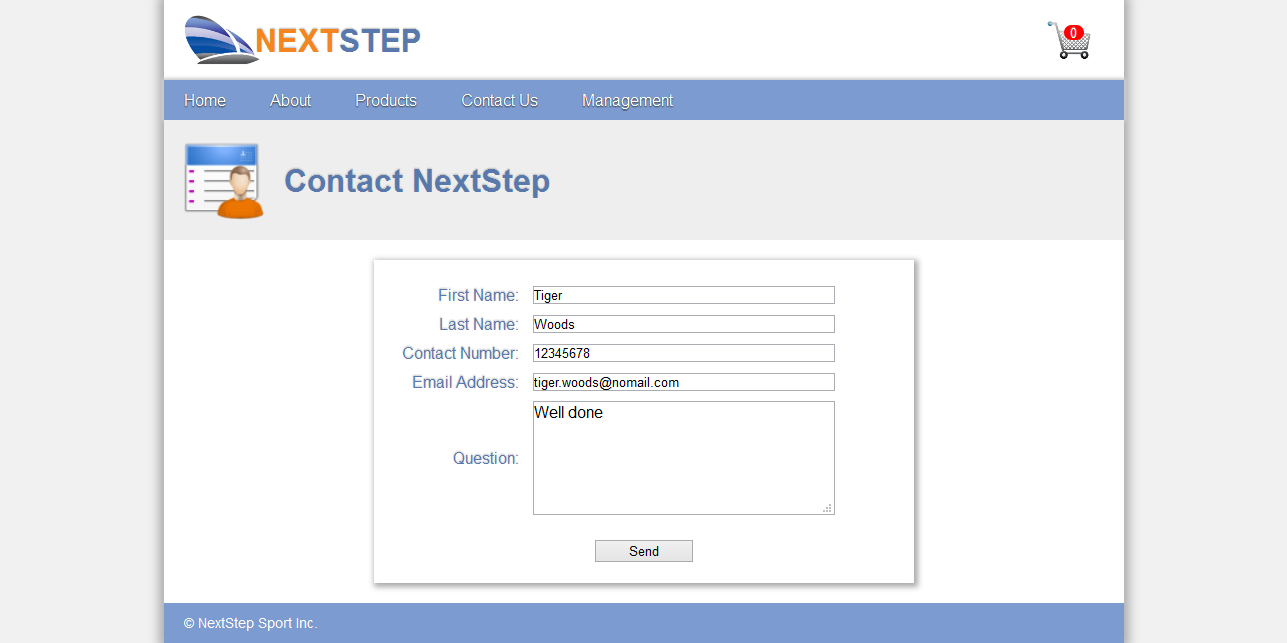
Navigation the major parts of NEXTSTEP website.



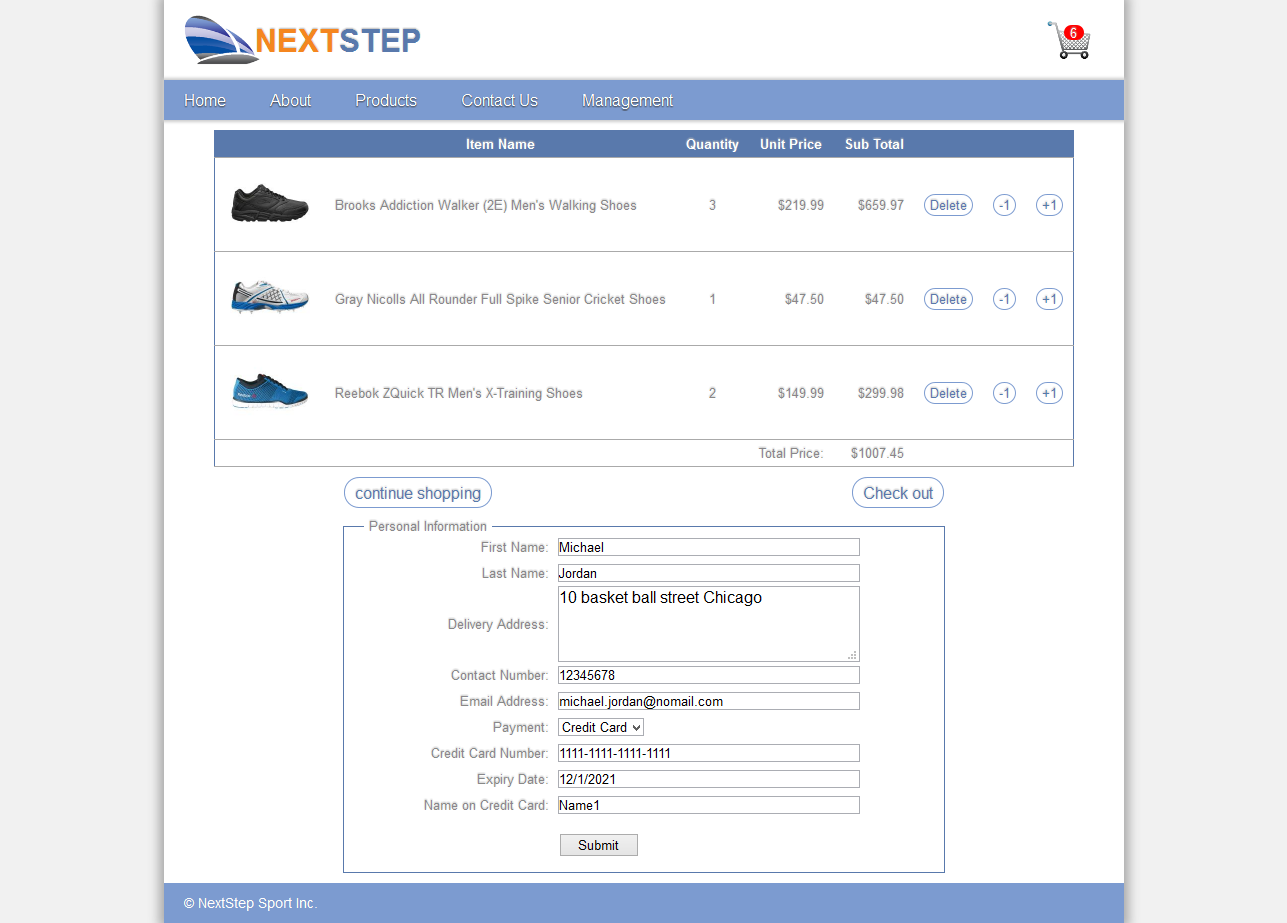




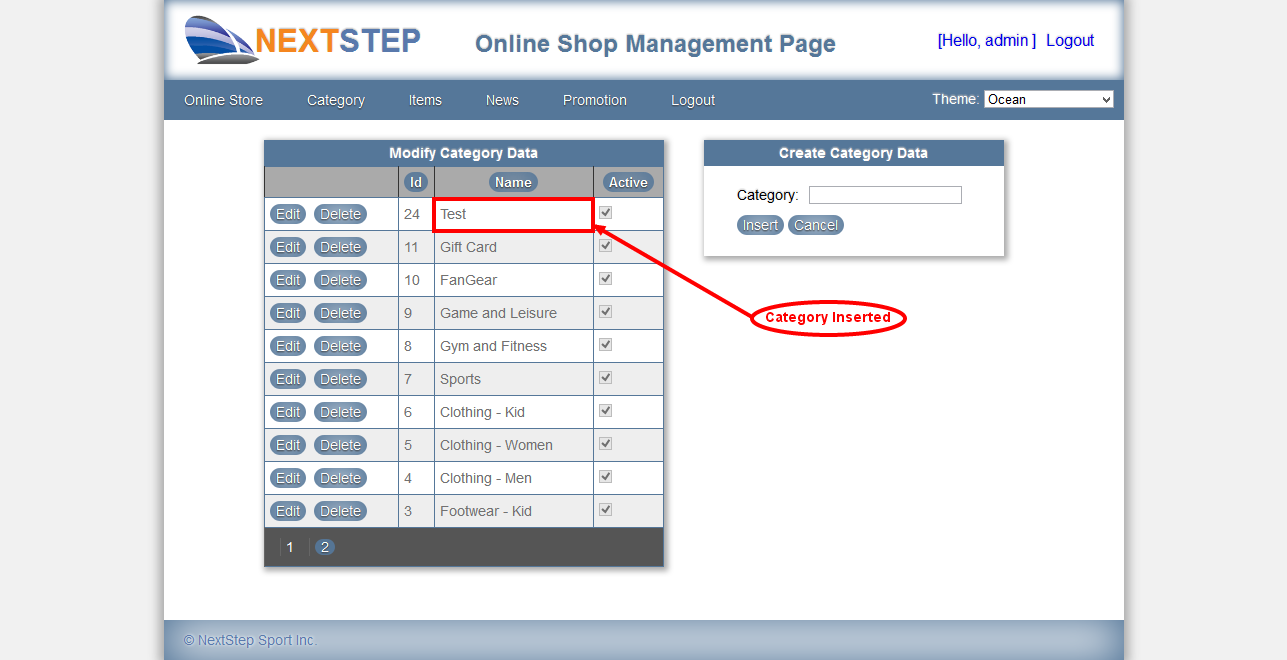
Contact Form



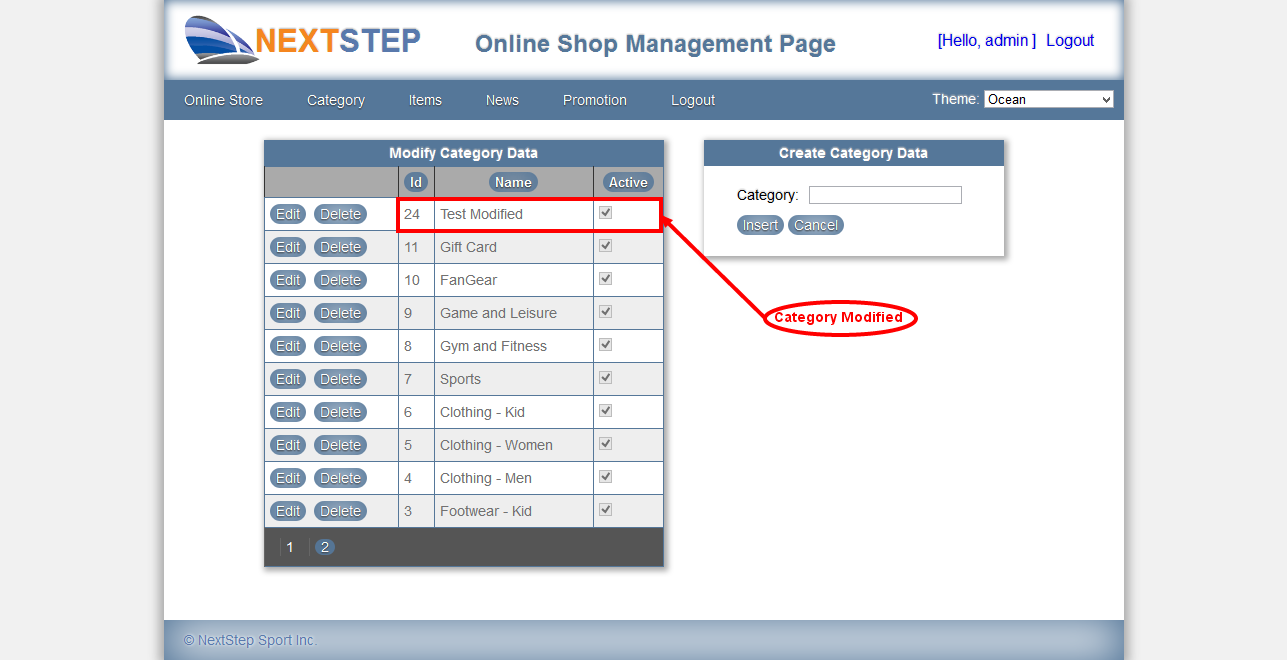
Shopping Cart - View shopping cart, edit cart content and checkout



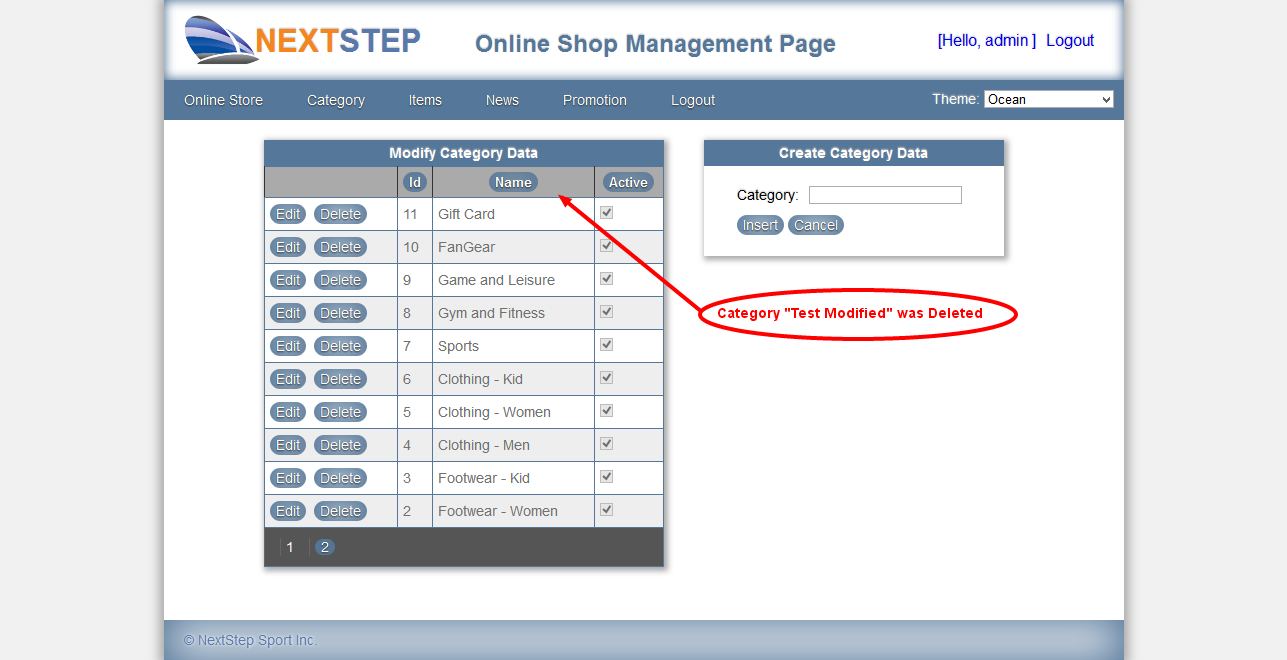
Insert Category



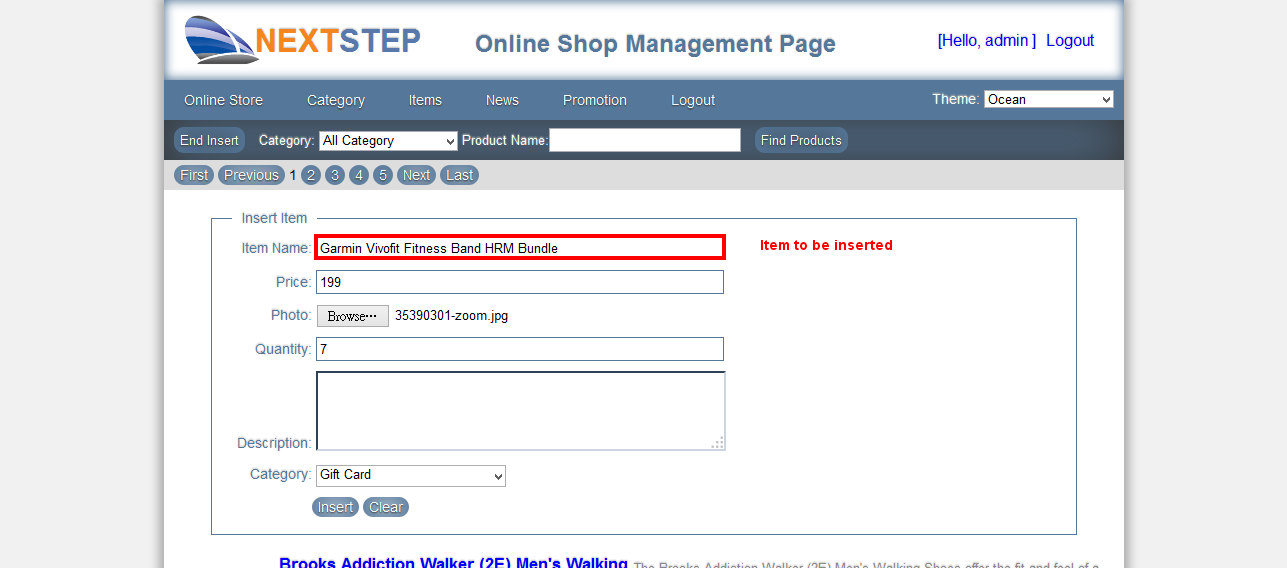
Edit Category

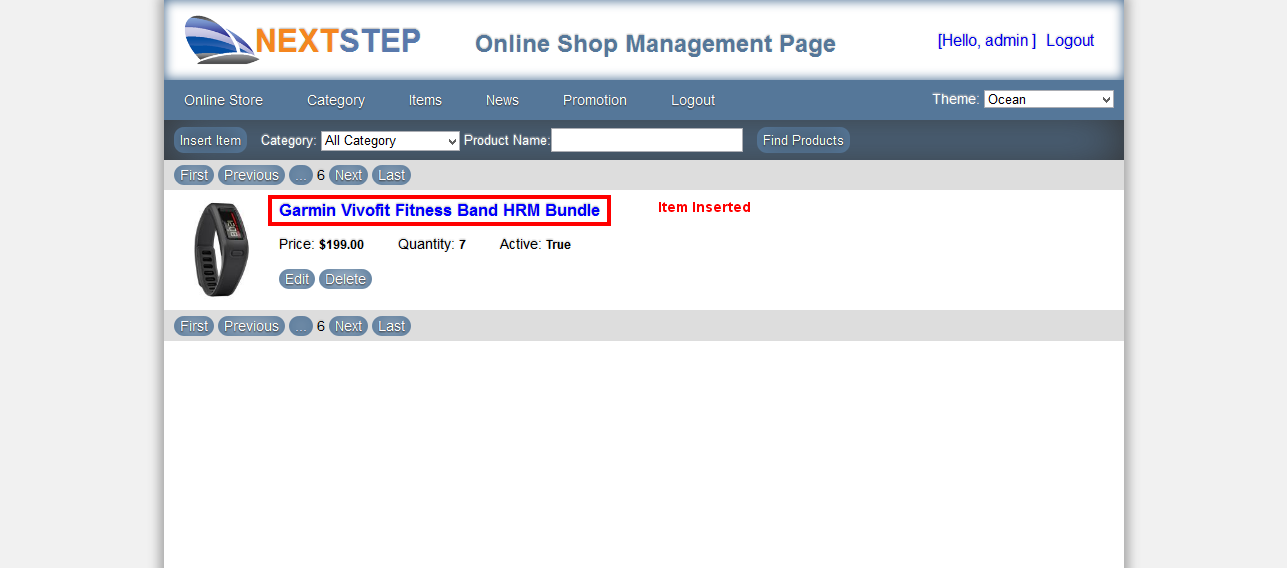


Delete Category

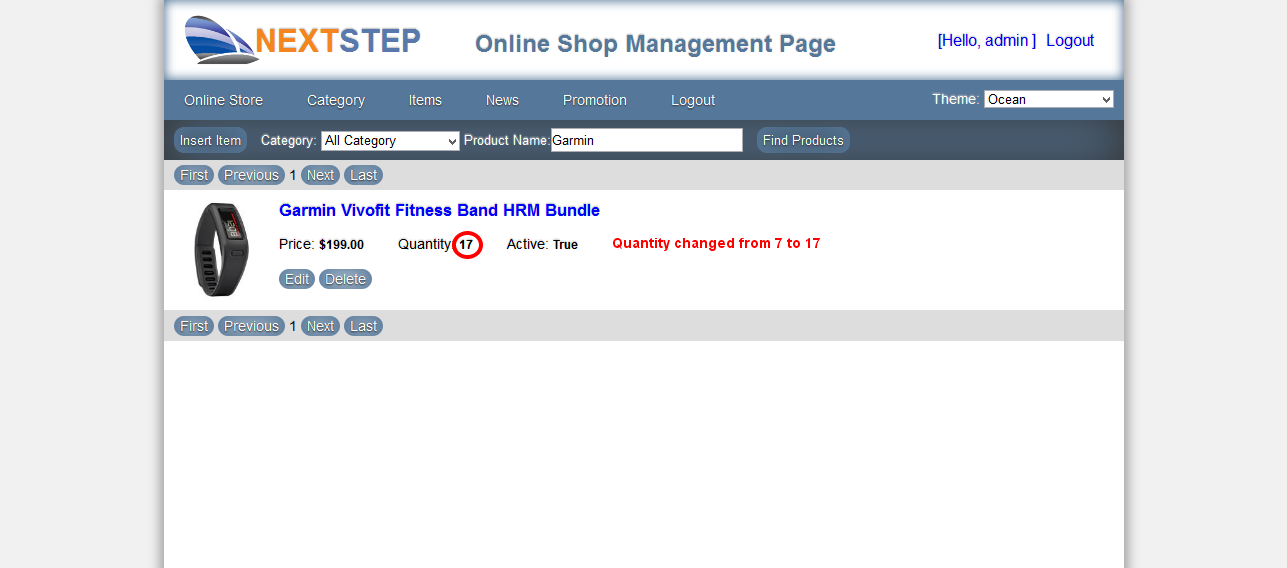


Insert Item

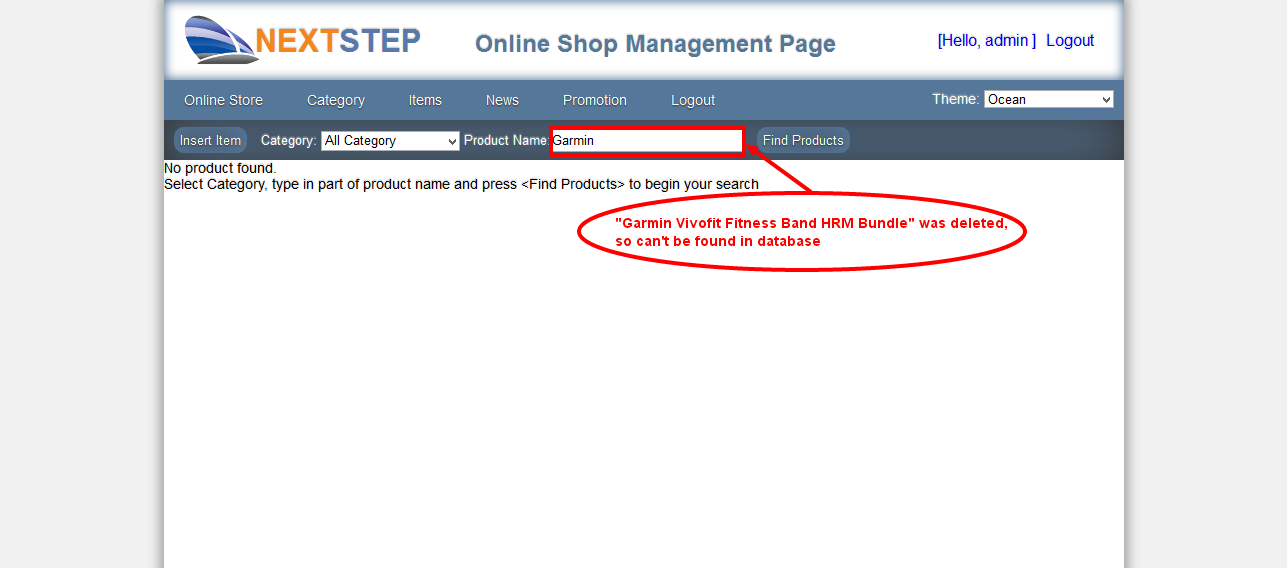




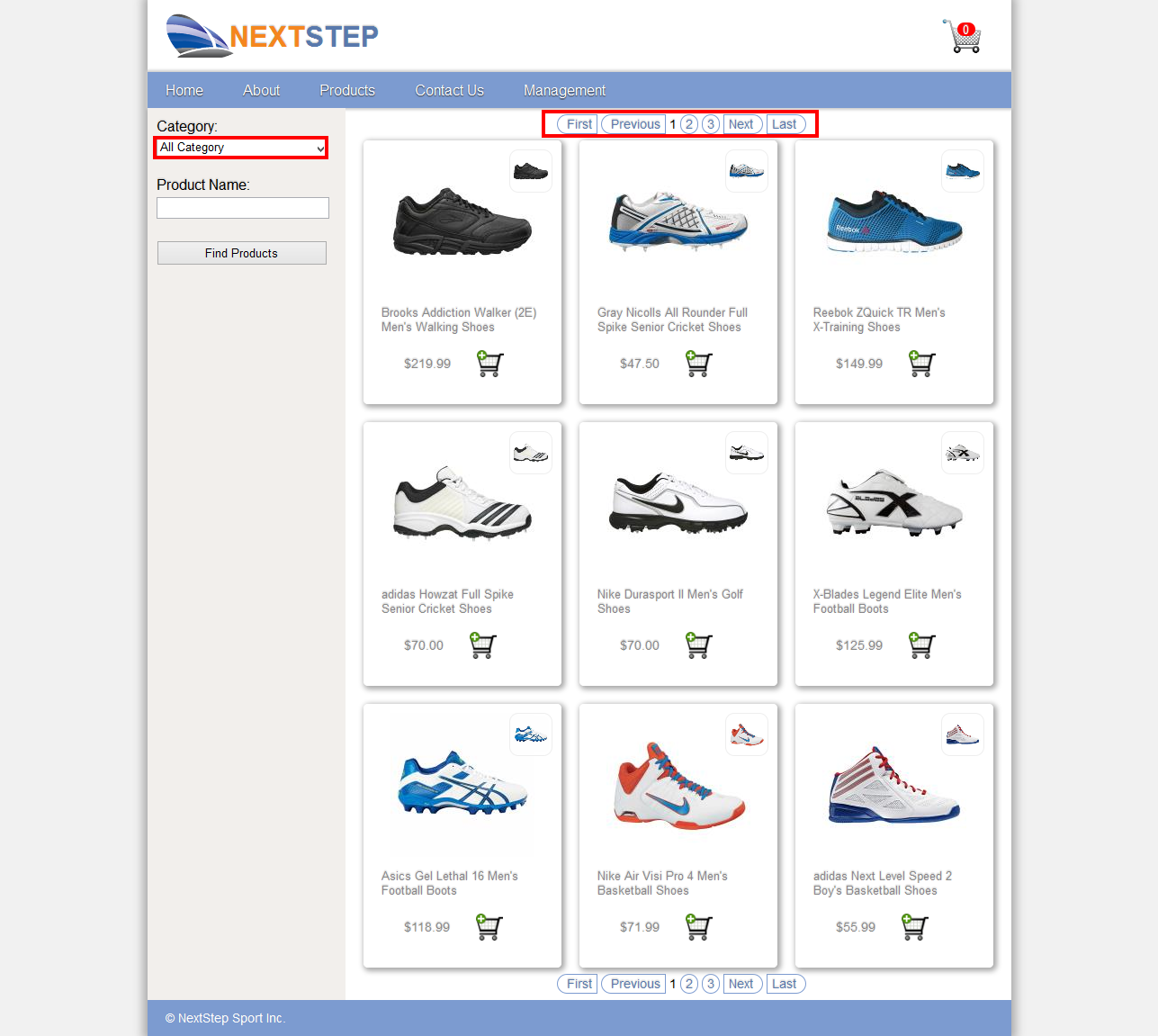
Edit Item



Delete Item



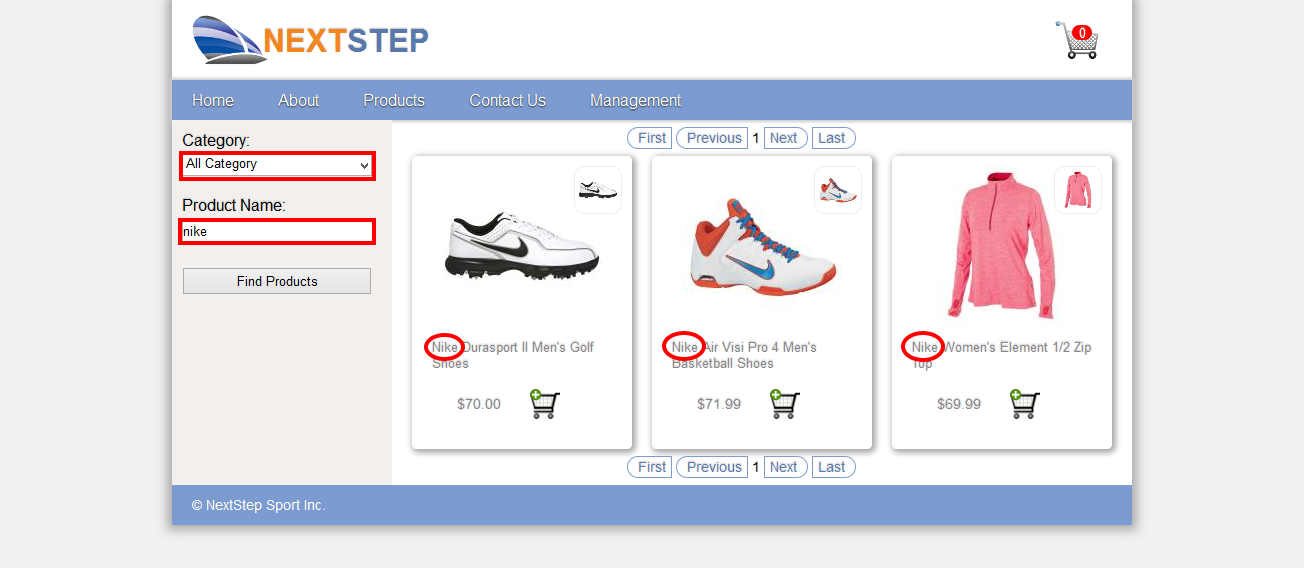
Find Products - find all products



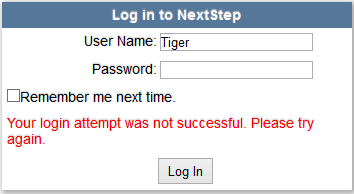
Find Products - by category



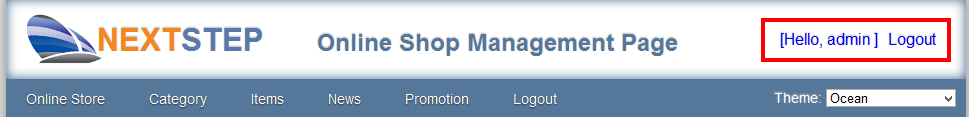
Find Products - with product name and category



Login - Fail login



Login - Successful login



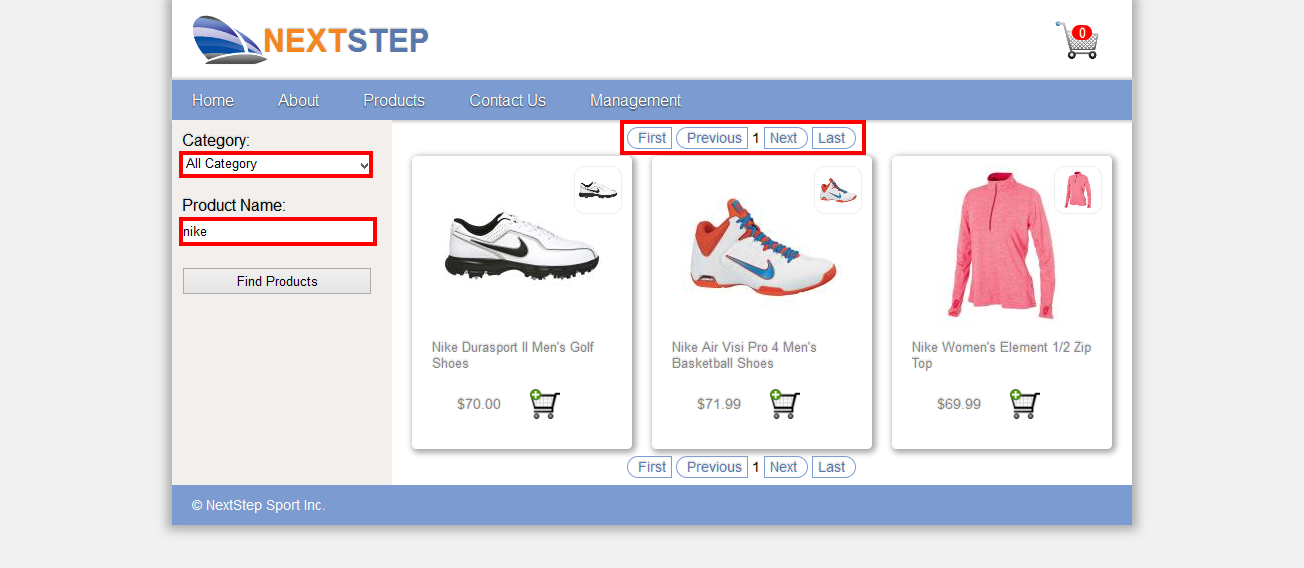
## Security Testing

A1 – Injection

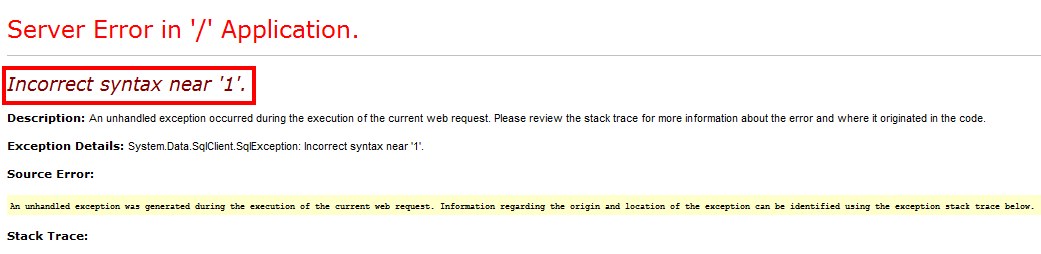
Injection flaws occur when an application sends untrusted data to an interpreter. Injection flaws are very prevalent, particularly in legacy code. They are often found in SQL, LDAP, Xpath, or NoSQL queries; OS commands; XML parsers, SMTP Headers, program arguments, etc. Injection flaws are easy to discover when examining code, but frequently hard to discover via testing. Scanners and fuzzers can help attackers find injection flaws

**V2-1 Product search with product name and all categories can be injected**

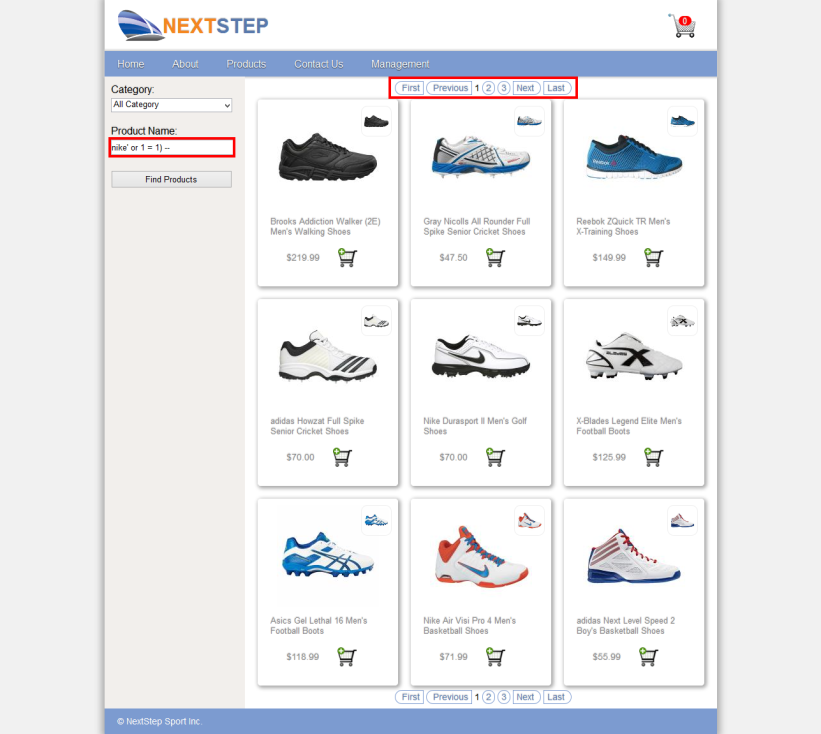
In the products page, when we select “All Category” and put “nike” into product name, we got 3 products as following picture.

****

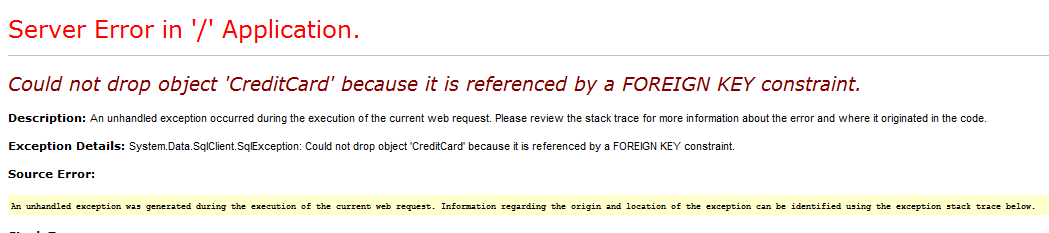
Try put “nike' or 1 = 1 – “ into product name field, we got an error message. The message tell us, if we put the right statement, it is going to accept our command.



Try put “nike' or 1 = 1) – “ into product name field again. This time we got all products returned.



Try put “nike' or 1 = 1) DROP TABLE CreditCard -- “ into product name field again. This time we got an error message “Could not drop object 'CreditCard' because it is referenced by a FOREIGN KEY constraint.” This message prove that the DROP command works.



**R2-1 Remedial action**

Solutions to the SQL injection in ASP.NET are

**R2-1-1 Validate the product name field**

You can limit the product name length or use regular expression to accept valid product name.

**R2-1-2 Using parameters for user input**

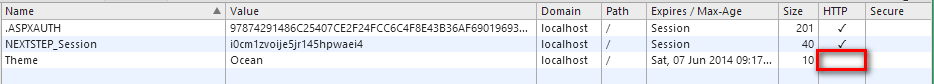
Using string concatenation is dangerous for SQL command preparation. Use parameter instead, that will help you get rid of the SQL injection mare.

A2 – Broken Authentication and Session Management

Developers frequently build custom authentication and session management schemes, but building these correctly is hard. As a result, these custom schemes frequently have flaws in areas such as logout, password management, timeouts, remember me, secret question, account update, etc. Finding such flaws can sometimes be difficult, as each implementation is unique.

**V2-1 Cookies are not http only by default**

The cookies of asp.net are not set to http only by default. That leads to easy reveal our personal information or credentials to hackers.



**R2-1 Remedial action**

By adding the httpOnlyCookies=”true” to web.config, we can ensure all cookies are http only. (You may delete old cookie first to see the change)

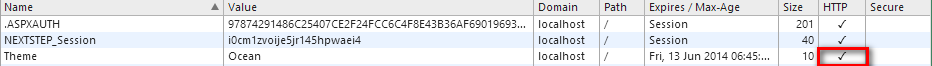
<configuration>

<system.webServer>

<httpCookies httpOnlyCookies="true"/>

</system.webServer>

</configuration>

****

**V2-2 Cookies are not secure by default**

The cookies of asp.net are not set to secure by default. We can see the Secure columns in the previous picture are not checked.

Authentication timeout

**R2-2 Remedial action**

By adding the requireSSL =”true” to web.config, we can ensure all cookies are secure.

<configuration>

<system.webServer>

<httpCookies httpOnlyCookies="true"

requireSSL="true"/>

</system.webServer>

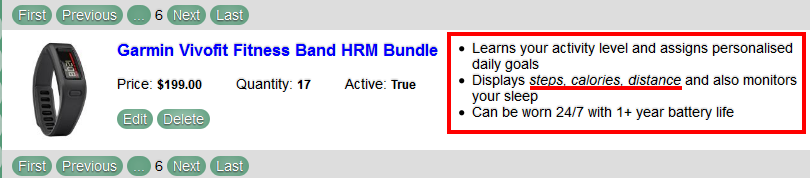
</configuration>

A3 – Cross-Site Scripting (XSS)

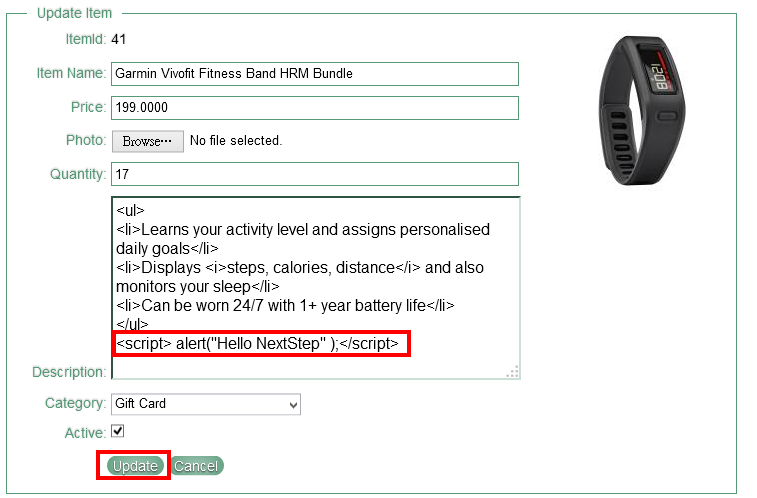
XSS is the most prevalent web application security flaw. XSS flaws occur when an application includes user supplied data in a page sent to the browser without properly validating or escaping that content. There are three known types of XSS flaws: 1) Stored, 2) Reflected, and 3) DOM based XSS. Detection of most XSS flaws is fairly easy via testing or code analysis.

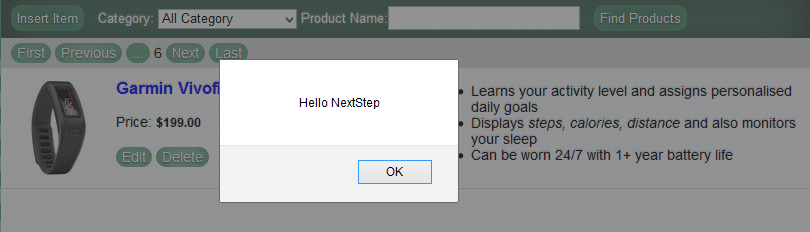
**V3-1 HTML tags in the description field**

In the management page, you can see the product description include some HTML. There may have some chance for XSS attack.



Try Edit the product, and append with <script> alert(“Hello NextStep” );</script>, click update button. You can see the alert window pops up, so you can put any javascript program here.





**R3-1 Remedial action**

There are some solutions for this case

**R3-1-1 Use plain text only**

The easiest way is to accept plain text only, just set the ValidateRequestMode property to enable. The default value of ValidateRequestMode is enable. To accept HTML tags, developer turn it into disable.

**R3-1-2 Use white list sanitization**

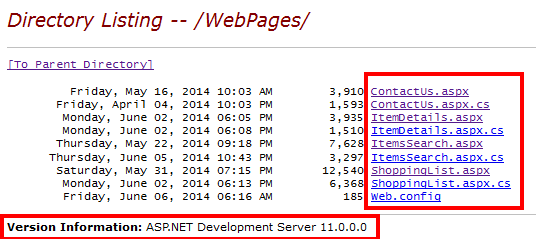
Sanitize all HTML tags except some healthy and needed tags, like <ul>, <li>, <b>, <i>, <p>.

A4 – Insecure Direct Object References

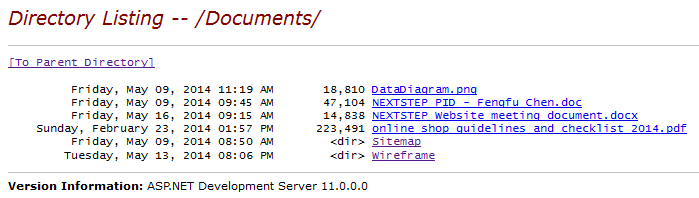
A direct object reference occurs when a developer exposes a reference to an internal implementation object, such as a file, directory, or database key. Without an access control check or other protection, attackers can manipulate these references to access unauthorized data.

**V4-1 Directories and files are public**

The URL of NEXTSTEP website’s products search page in our test environment is <http://localhost:1490/WebPages/ItemsSearch.aspx>. That looks normal, but it tell us something there too. “There is a folder called WebPages”. So let’s try <http://localhost:1490/WebPages/>. You can see the all the files in WebPages folder and the development environment with its version information. Try right-click over any file, like contact.aspx.cs, and then select “save link as”, and then select a folder to save it. You can download any file in this folder without any permission.



Try another URL <http://localhost:1490/Documents/>, and you can see following



The website developer put all their development documents here.

**R4-1 Remedial action**

There are several ways to protect unwanted directory listing

**R4-1-1 Change web.config at website root directory**

**R4-1-1-1 Disable Directory Browse**

<configuration>

<system.webServer>

<directoryBrowse enabled="false" />

</system.webServer>

</configuration>

**R4-1-1-1 Disable all directories browse**

<configuration>

<system.webServer>

<directoryBrowse enabled="false" />

</system.webServer>

</configuration>

**R4-1-1-2 Allow all directories browse for authorized users only**

<configuration>

<system.web>

<authorization>

<deny users="?"/>

</authorization>

</system.web>

</configuration>

**R4-1-2 Change web.config at specific sub-directory**

**R4-1-2-1 Disable specific directory browse**

<configuration>

<system.webServer>

<directoryBrowse enabled="false" />

</system.webServer>

</configuration>

**R4-1-1-2 Allow specific directory browse for authorized users only**

<configuration>

<system.web>

<authorization>

<deny users="?"/>

</authorization>

</system.web>

</configuration>

The default value of directoryBrowse in IIS 8.0 is

<directoryBrowse enabled="true" />

The default value of directoryBrowse in Visual Studio 2012 Development environment is

<directoryBrowse enabled="true" />

Don’t rely on the default setting. The settings could be different for different version. You should set the value by yourself to have a consistent result for different environment.

**R4-1-3 Add default.aspx to specific folder**

When users requested a directory, ASP.NET find default.aspx file in this directory first. If default.asp found, it will be send to user’s browser. So we can use default.aspx for protection.

**R4-1-3-1 Display an error message**

Include some message in the default.aspx, like

<h1>Directory browse are not allowed in the site</h1>

**R4-1-3-1 Redirect to other page**

Include a redirect command in the default.aspx, like

Response.Redirect("~/Default.aspx");

All requests to this directory will be redirect to website’s home page.

**R4-1-4 Remove unnecessary data from website**

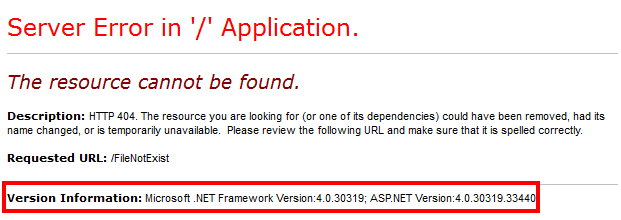
It’s the first policy for website security, because no website is really safe.

A5 – Security Misconfiguration

Security misconfiguration can happen at any level of an application stack, including the platform, web server, application server, database, framework, and custom code. Developers and system administrators need to work together to ensure that the entire stack is configured properly. Automated scanners are useful for detecting missing patches, misconfigurations, use of default accounts, unnecessary services, etc. Additionally, software should be kept up to date.

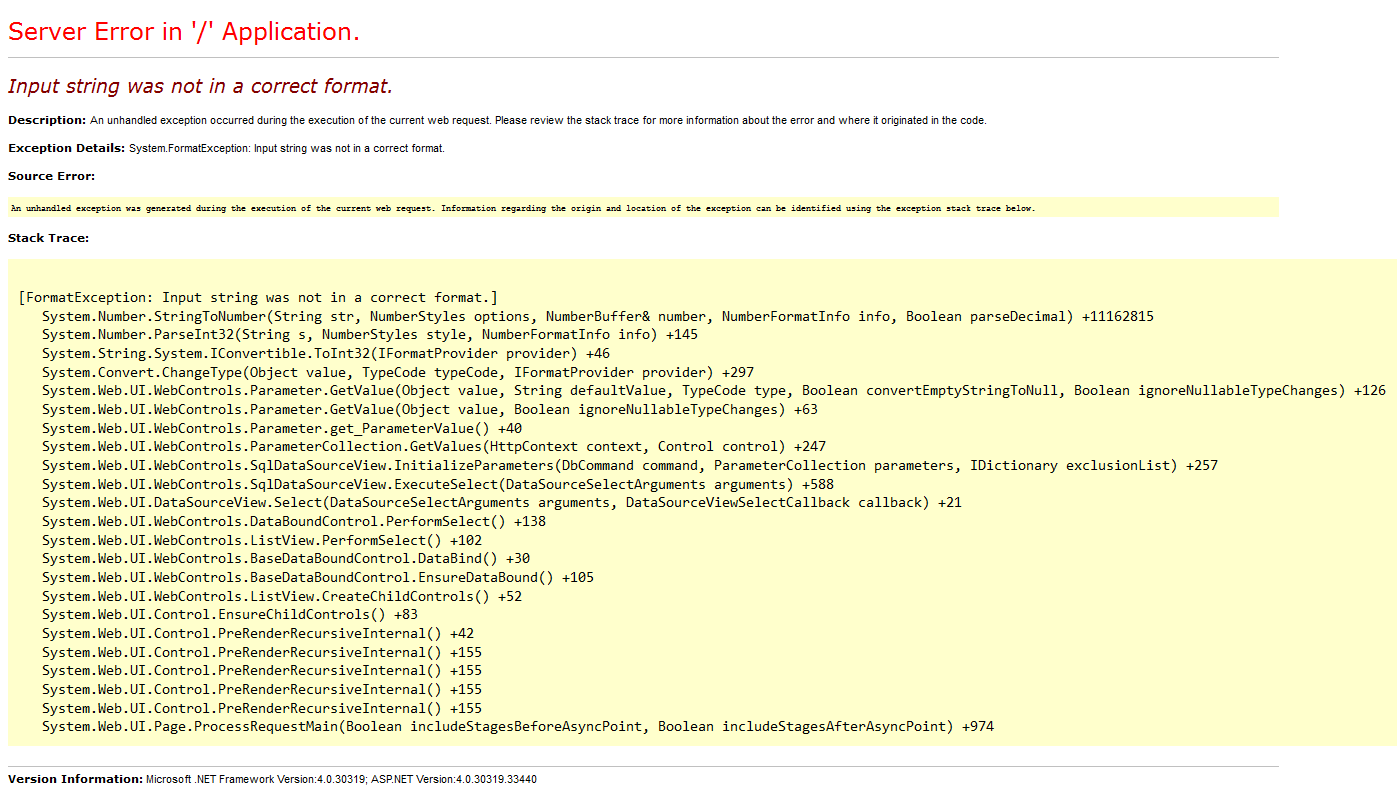
**V5-1 Error messages reveal internal information**

When we request a non-exist page, say <http://localhost:1490/FileNotExist>, the webserver return error message as following.

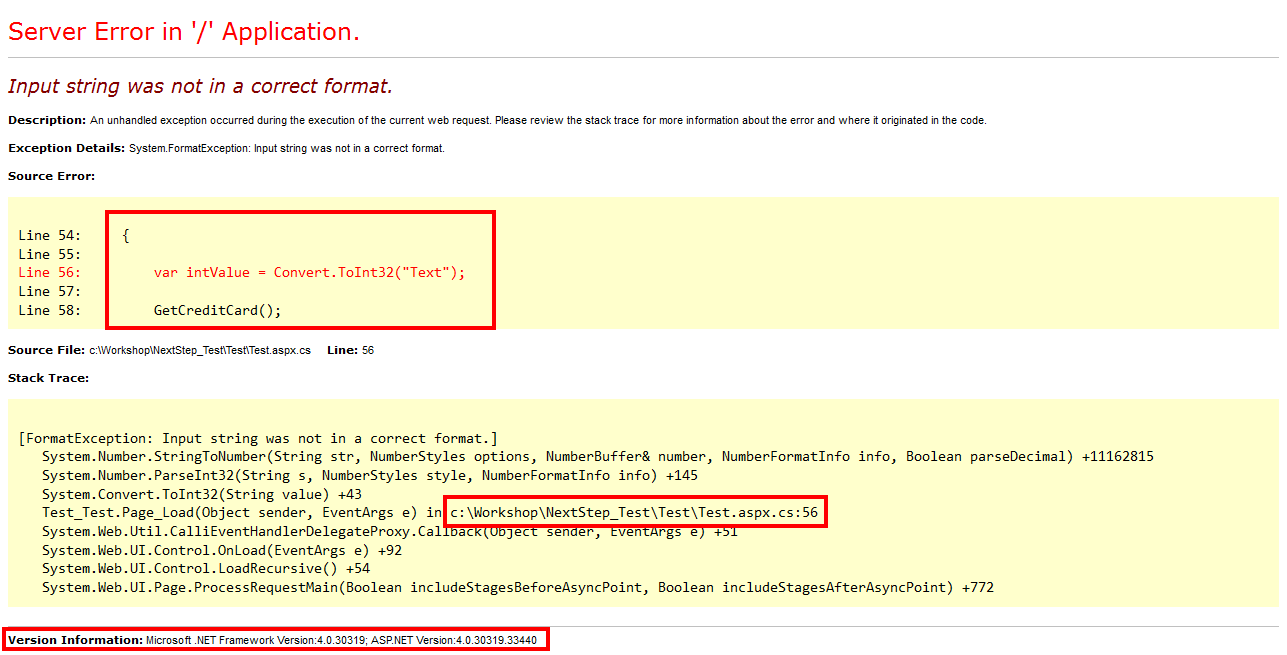


The message reveals your development environment to user.

If we try to make some error requests for the server, like put a text to an integer field <http://localhost:1490/WebPages/ItemDetails.aspx?Id=text>, we will get following message back.



This message reveals more information to user. If our code has any uncaught error. The server will response even more information to user, include our code, internal path and development environment.



**R5-1 Remedial action**

The solution to this kind of error messages is to enable customer errors in web.config file.

<configuration>

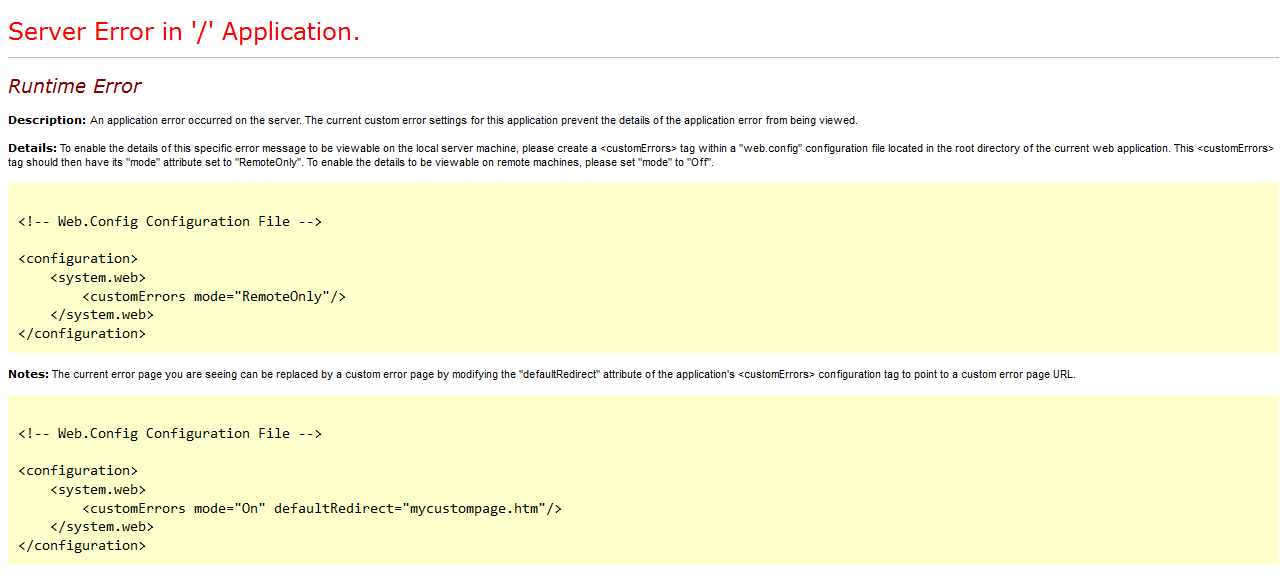
<system.web>

<customErrors mode="On"/>

</system.web>

</configuration>

Then the message will change to



That’s fine. We secure the error message, but as a developer, I want more information for debug, that messages can’t help. A better way is set custom errors mode to “RemoteOnly”, and if you want provide a more meaningful message to user. You can assign a default redirect page. The final solution is following.

<configuration>

<system.web>

<customErrors mode="RemoteOnly" defaultRedirect="Errors.aspx"

redirectMode="ResponseRedirect"/>

</system.web>

</configuration>

Other ASP.NET security issues

**B1 Trace option**

Tracing is a very handy feature for spitting out diagnostic information about what's going on inside the web application. We can turn tracing on just by jumping down web.config, adding a trace element, and turning it on.

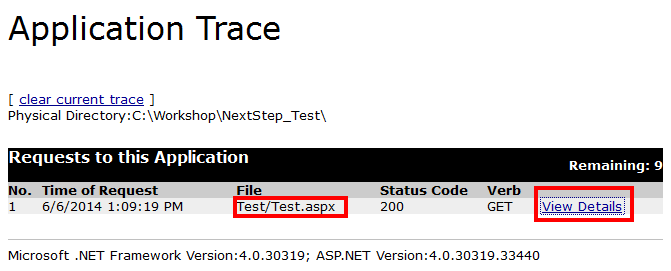
<configuration>

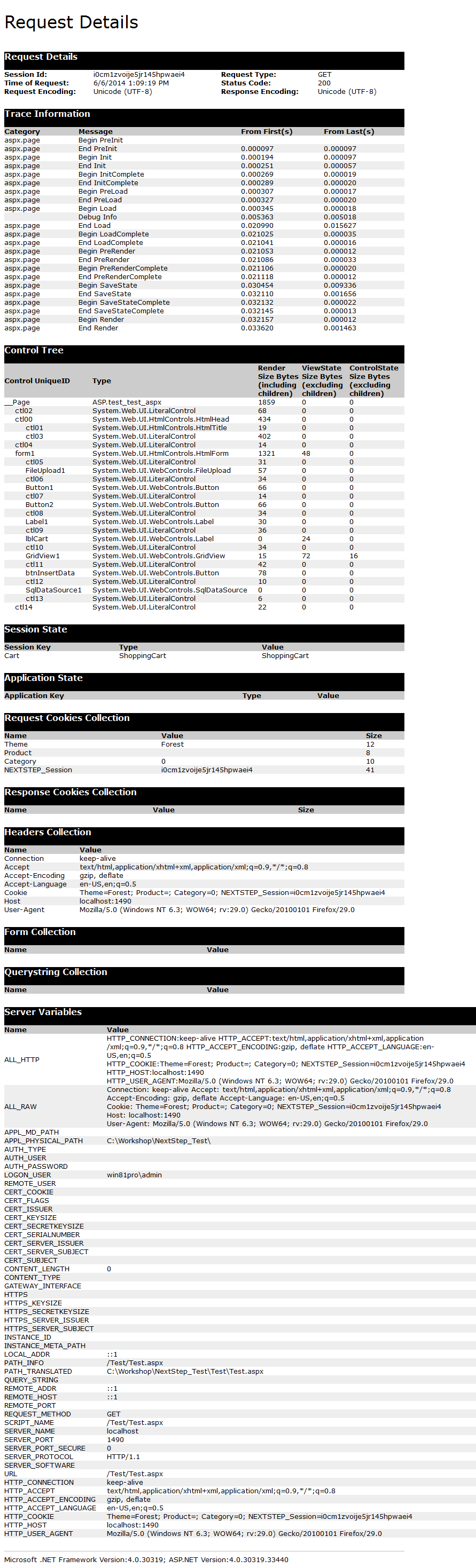
<system.web>

<trace enabled="true"/>

</system.web>

</configuration>

Let‘s put <http://localhost:1490/trace.axd> into browser, and see what happen.We can see that developer did a Trace.write(“”); statement in Test.aspx, and a View Details link. Click the link. We get lots of information about our website, including Request Details, Trace Information, Control Tree, Session State, Application State, Request Cookies Collection, Response Cookies Collection, Headers Collection, Form Collection, Query Collection and Server Variables.



**B1 Trace option - Solution**

Always disable trace option at production site. When you turn on the trace option, it’s a good habit to set localOnly property to true.

<configuration>

<system.web>

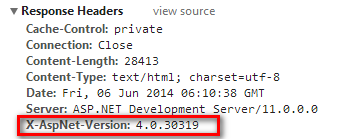
<trace enabled="true" localOnly="true" />

</system.web>

</configuration>

**B2 Http Header information**

Asp.net reveals some information for every request in the http response header, including Server and development tool.

****

**B2 Http Header information**

We can tell asp.net not to reveal our development tool by changing the web.config

<configuration>

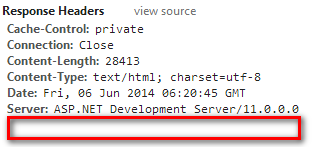
<system.web>

<httpRuntime targetFramework="4.5"

enableVersionHeader="false">

</system.web>

</configuration>

****

## appendix

Selenium Scripts for this testing

| Contact | | |
| --- | --- | --- |
| open | / |  |
| clickAndWait | link=Contact Us |  |
| type | id=ContentPlaceHolder1\_tbFirstName | Tiger |
| type | id=ContentPlaceHolder1\_tbLastName | Woods |
| type | id=ContentPlaceHolder1\_tbContactNumber | 12345678 |
| type | id=ContentPlaceHolder1\_tbEmailAddress | tiger.woods@nomail.com |
| type | id=ContentPlaceHolder1\_tbQuestion | Well done |
| clickAndWait | id=ContentPlaceHolder1\_btnSubmit |  |

| Delete Category | | |
| --- | --- | --- |
| open | /Management/CategoryManagement.aspx |  |
| clickAndWait | link=Online Store |  |
| clickAndWait | link=Management |  |
| clickAndWait | link=Category |  |
| click | id=ContentPlaceHolder1\_GridView1\_LinkButton2\_0 |  |
| assertConfirmation | Are you sure you want to delete this Category? |  |

| DeleteItem | | |
| --- | --- | --- |
| open | / |  |
| clickAndWait | link=Management |  |
| clickAndWait | link=Items |  |
| select | id=ContentPlaceHolder1\_ddlCategory | label=All Category |
| type | id=ContentPlaceHolder1\_txtProduct | Garmin |
| clickAndWait | id=ContentPlaceHolder1\_btnFindProduct |  |
| type | id=ContentPlaceHolder1\_txtProduct | Garmin |
| clickAndWait | id=ContentPlaceHolder1\_btnFindProduct |  |
| click | id=ContentPlaceHolder1\_ListView1\_DeleteButton\_0 |  |
| assertConfirmation | Are you sure you want to delete this product? |  |

| EditCategory | | |
| --- | --- | --- |
| open | / |  |
| clickAndWait | link=Management |  |
| clickAndWait | link=Category |  |
| clickAndWait | id=ContentPlaceHolder1\_GridView1\_LinkButton1\_0 |  |
| type | name=ctl00$ContentPlaceHolder1$GridView1$ctl02$ctl00 | Test Modified |
| clickAndWait | id=ContentPlaceHolder1\_GridView1\_LinkButton1\_0 |  |

| EditItem | | | | | |
| --- | --- | --- | --- | --- | --- |
| open | | / | | |  |
| clickAndWait | | link=Management | | |  |
| clickAndWait | | link=Items | | |  |
| type | | id=ContentPlaceHolder1\_txtProduct | | | Garmin |
| clickAndWait | | id=ContentPlaceHolder1\_btnFindProduct | | |  |
| clickAndWait | | id=ContentPlaceHolder1\_ListView1\_EditButton\_0 | | |  |
| type | | id=ContentPlaceHolder1\_ListView1\_QuantityTextBox\_0 | | | 17 |
| clickAndWait | | id=ContentPlaceHolder1\_ListView1\_UpdateButton\_0 | | |  |
| assertText | | //div[@id='ContentPlaceHolder1\_ListView1\_itemPlaceholderContainer']/div/div/h5[2] | | | 17 |
| FindProducts - All | | | |
| open | /WebPages/ItemsSearch.aspx | |  |
| type | id=ContentPlaceHolder1\_txtProduct | |  |
| clickAndWait | id=ContentPlaceHolder1\_btnFindProduct | |  |

| FindProducts - Category | | |
| --- | --- | --- |
| open | /WebPages/ItemsSearch.aspx |  |
| clickAndWait | link=Products |  |
| select | id=ContentPlaceHolder1\_ddlCategory | label=Footwear - Men |
| clickAndWait | id=ContentPlaceHolder1\_btnFindProduct |  |
| select | id=ContentPlaceHolder1\_ddlCategory | label=Footwear - Women |
| clickAndWait | id=ContentPlaceHolder1\_btnFindProduct |  |
| select | id=ContentPlaceHolder1\_ddlCategory | label=Clothing - Kid |
| clickAndWait | id=ContentPlaceHolder1\_btnFindProduct |  |

| FindProducts - Nike | | |
| --- | --- | --- |
| open | /WebPages/ItemsSearch.aspx |  |
| clickAndWait | link=Products |  |
| type | id=ContentPlaceHolder1\_txtProduct | nike |
| clickAndWait | id=ContentPlaceHolder1\_btnFindProduct |  |
| assertText | id=ContentPlaceHolder1\_ListView1\_ItemNameLabel\_0 | Nike Free Trainer 5.0 Men's Running Shoes |

| Insert Category | | |
| --- | --- | --- |
| open | /Management/Management.aspx |  |
| clickAndWait | css=img.logo |  |
| clickAndWait | link=Management |  |
| clickAndWait | link=Category |  |
| type | id=ContentPlaceHolder1\_DetailsView1\_txtCategory | Test |
| clickAndWait | link=Insert |  |
| assertText | //table[@id='ContentPlaceHolder1\_GridView1']/tbody/tr[2]/td[3] | Test |

| InsertItem | | |
| --- | --- | --- |
| open | / |  |
| clickAndWait | link=Management |  |
| clickAndWait | link=Items |  |
| clickAndWait | id=ContentPlaceHolder1\_btnInsert |  |
| type | id=ContentPlaceHolder1\_ListView1\_ItemNameTextBox | Garmin Vivofit Fitness Band HRM Bundle |
| type | id=ContentPlaceHolder1\_ListView1\_PriceTextBox | 199 |
| type | id=ContentPlaceHolder1\_ListView1\_QuantityTextBox | 7 |
| type | id=ContentPlaceHolder1\_ListView1\_FileUpload1 | C:\Temp\35390301-zoom.jpg |
| select | id=ContentPlaceHolder1\_ListView1\_ddlCategoryEdit | label=Gift Card |
| clickAndWait | id=ContentPlaceHolder1\_ListView1\_InsertButton |  |
| clickAndWait | id=ContentPlaceHolder1\_btnInsert |  |
| clickAndWait | link=Last |  |

| Login-Fail | | |
| --- | --- | --- |
| open | / |  |
| clickAndWait | link=Management |  |
| type | id=ContentPlaceHolder1\_Login1\_UserName | Tiger |
| type | id=ContentPlaceHolder1\_Login1\_Password | 12345678 |
| clickAndWait | id=ContentPlaceHolder1\_Login1\_LoginButton |  |
| assertText | css=p.error | Your login attempt was not successful. Please try again. |

| Login-Success | | |
| --- | --- | --- |
| open | /Management/Login.aspx?ReturnUrl=%2fManagement%2fManagement.aspx |  |
| clickAndWait | css=img.logo |  |
| clickAndWait | link=Management |  |
| type | id=ContentPlaceHolder1\_Login1\_UserName | admin |
| type | id=ContentPlaceHolder1\_Login1\_Password | password! |
| clickAndWait | id=ContentPlaceHolder1\_Login1\_LoginButton |  |
| assertText | id=LoginStatus2 | Logout |

| Navigation | | |
| --- | --- | --- |
| open | /Management/Login.aspx?ReturnUrl=%2fManagement%2fManagement.aspx |  |
| clickAndWait | css=img.logo |  |
| clickAndWait | link=About |  |
| clickAndWait | link=Products |  |
| clickAndWait | link=Contact Us |  |
| clickAndWait | link=Management |  |

| ShoppingCart | | |
| --- | --- | --- |
| open | / |  |
| clickAndWait | css=img.logo |  |
| clickAndWait | link=Products |  |
| select | id=ContentPlaceHolder1\_ddlCategory | label=All Category |
| clickAndWait | id=ContentPlaceHolder1\_btnFindProduct |  |
| clickAndWait | id=ContentPlaceHolder1\_ListView1\_imgAddToCart\_0 |  |
| clickAndWait | id=ContentPlaceHolder1\_ListView1\_imgAddToCart\_1 |  |
| clickAndWait | id=ContentPlaceHolder1\_ListView1\_imgAddToCart\_2 |  |
| clickAndWait | id=lblNumProducts |  |
| clickAndWait | id=ContentPlaceHolder1\_GridView1\_lbPlusOne\_0 |  |
| clickAndWait | id=ContentPlaceHolder1\_GridView1\_lbPlusOne\_0 |  |
| clickAndWait | id=ContentPlaceHolder1\_GridView1\_lbPlusOne\_1 |  |
| clickAndWait | id=ContentPlaceHolder1\_GridView1\_lbPlusOne\_2 |  |
| clickAndWait | id=ContentPlaceHolder1\_GridView1\_lbMinusOne\_1 |  |
| clickAndWait | id=ContentPlaceHolder1\_lbCheckOut |  |
| type | id=ContentPlaceHolder1\_tbFirstName | Tommy |
| type | id=ContentPlaceHolder1\_tbLastName | Jones |
| type | id=ContentPlaceHolder1\_tbFirstName | Michael |
| type | id=ContentPlaceHolder1\_tbLastName | Jordan |
| type | id=ContentPlaceHolder1\_tbAddress | 10 basket ball street Chicago |
| type | id=ContentPlaceHolder1\_tbContactNumber | 12345678 |
| type | id=ContentPlaceHolder1\_tbEmailAddress | michael.jordan@nomail.com |
| type | id=ContentPlaceHolder1\_tbCreditCard | 1111-1111-1111-1111 |
| type | id=ContentPlaceHolder1\_tbExpiry | 12/1/2021 |
| type | id=ContentPlaceHolder1\_tbNameOnCard | Name1 |
| clickAndWait | id=ContentPlaceHolder1\_btnSubmit |  |