WHCK Web Beta v1.0

Contents

[Document Last Update: 1](#_Toc340848962)

[Contact Info: 1](#_Toc340848963)

[Tsiewhckweb TestResult Management Web Application – Version 1.0 Signoff Criteria (Beta Release v1.0) 1](#_Toc340848964)

[Apply the following minimum required fuctional test: 1](#_Toc340848965)

[“ICS” Setup Guide for WHCK Web TestResult Management Web Application (that uses .hckx file package processing): 2](#_Toc340848966)

[Steps to Install SQL Server: 2](#_Toc340848967)

[Steps to Setup Windows Firewall Open access to SQL Server: 2](#_Toc340848968)

[Steps to Install IIS Web Server 3](#_Toc340848969)

[Steps to Publish and Compile WHCK Web+Database 3](#_Toc340848970)

# Document Last Update:

November 16th, 2012

# Contact Info:

Manager: Solomon Kinard email: [solomon.kinard@intel.com](mailto:solomon.kinard@intel.com)

Test: Himanshu Pathak email: [himanshux.pathak@intel.com](mailto:himanshux.pathak@intel.com)

Development/Test: Matthew James Klump email: [matthewx.j.klump@intel.com](mailto:matthewx.j.klump@intel.com)

# Tsiewhckweb TestResult Management Web Application – Version 1.0 Signoff Criteria (Beta Release v1.0)

## Apply the following minimum required fuctional test:

1. Step 1: Go to the officially designated web application hosting site.
2. Step 2: Add 3 to 10 Valid test cases using the TestCase management link from the Home page.
3. Step 3: Add 3 to 5 Valid WHCK .hckx extention test result packages presently required for processing by using the Configuration link on the Home page.
4. Step 4: Click the Configuration (~/Configuration/) View page from on the Home page for correctness of each test config saved with each test result package for correctness.
5. Step 5: Click each details link (~/Configuration/details) from the Home page, and examine the data about the package that is being saved there for each test config and package data that is saved there for correctness.
6. Step 6: Click the Testing Status link (~/TestingStatus/) link from on the Home page, and examine the test passing, and the test failing average percentage calculations for correctness based on exactly what was entered especially in steps 1 and 2.

# “ICS” Setup Guide for WHCK Web TestResult Management Web Application (that uses .hckx file package processing):

Step 1: Result of copying pre-built/compiled test contents 🡪 WARNING 🡨 Do Not Do This!

Please first follow steps first below under the section “Steps to Publish and Compile WHCK Web+Database”…

Step 2: The tsiewhckweb database is presently being hosted on SQL Server 2008 R2 on test machine server “tsiewhckweb1”, and needs to be moved also from the test hosting server to desired production server. This can be done be consulting SQL Books Online contents (<http://msdn.microsoft.com/en-us/library/ms130214(v=sql.100).aspx)> for deploying a new SQL Server instance and application database.

# Steps to Install SQL Server:

1. Run SQL Setup and add “SQL Server Engine”, “Client Connectivity” for a network conntection, “Integration Services” for the database transfer and backup tools, SQL Database Engine Server Core, and also Management Tools Basic+.
2. During SQL Server setup, please select to only install the default instance (local) instance and not a named instance, or the SQL Server will become more difficult to address as a named instnance (server\namedinstance) instead of (local).
3. Also during the SQL Server setup, for all services that local system and network access, set those to security access level SYSTEM account. For the service only requiring network access, please set those services to only security access level NETWORK SERVICE.
4. Please select MIXED MODE authentication, and set a SECURE password the “sa” admin super user account.
5. Finish the install wizard clicking next, and install begins to execute.

# Steps to Setup Windows Firewall Open access to SQL Server:

1. Right click the network icon that appears in the graphic here :
2. <image of network icon>
3. Select Network and Sharing Center.
4. In the lower left of that window, click on Windows Firewall.
5. Click on “Advanced settings”.
6. Click on “In bound Rules” and is selected under “Windows Firewall with Advanced Security under Local Computer” mmc administrative span-in.
7. In the right side action pain, click on “New Rule…” that will open the “New Inbound Rule Wizard”.
8. Click on the “Port” radio button.
9. For the port type, leave the selection as “TCP”, and enter the SQL engine TCP port exception 1433, and click next.
10. Leave the selection as “Allow the connection”, and click next.
11. Leave the current selection for all network profiles “Domain”, Private, and Public, and click next.
12. Enter the rule name as “SQL Server Database Engine network connection”, and click “Finish.”
13. Still in the same window click again “Advanced settings”, and follow the prior steps 8) through 15), and open the Windows Firewall to allow also connections to TCP port 1434 in addition to 1433, and also name the Inbound Rule as “SQL Server Database Engine network connection”.

# Steps to Install IIS Web Server

The application is presently being hosted using Microsoft IIS 7.5 web server. The application requires a web server such as Apache or IIS web servers to functions. Instructions for installing IIS web server can be found here -> (<http://www.iis.net/learn/install/installing-iis-7/installing-iis-7-and-above-on-windows-server-2008-or-windows-server-2008-r2>).

* 1. Launch Server Manager
  2. Click “Add Role”.
  3. Select IIS Web Server
  4. Select All required IIS components, and click to finish wizard and start Installing IIS Web Server.
  5. Install ASP.NET 4.0: dotNetFx40\_Full\_x86\_x64.exe which is the download version of .NET 4.0.
  6. Install MVC 4: AspNetMVC4Setup.exe download version of ASP.NET MVC Razor page/view rendering engine.
  7. Launch IIS Manager.
  8. In the left node tree, right click the node for “AppPool”, and click advanced.
  9. In the section for “Enable 32-bit Applications”, please set that to TRUE, and then click “OK”.
  10. Follow step for deploying a new virtual directory, and then setting that virtual directory as a web application. Instructions for doing this can be found here -> (<http://msdn.microsoft.com/en-us/library/bb763173(v=vs.90).aspx)>.

# Steps to Publish and Compile WHCK Web+Database

1. Set a folder level network share ([\\tsievauto1\whckweb](file:///\\tsievauto1\whckweb)) with write permissions sufficient for Visual Studio tool publish function directly from the build.
2. Copy unpublished dbo.tsiewhckweb\_CREATE\_DB.sql to the target server machine network share.
3. Open SQL Management Studio, connect to local running instance of SQL Server using SA account.
4. Right the Database node, and select new database.
5. Keep all the defaults, and enter the name “whckweb” for a database name.
6. At the top of the script, change “USE tsiewhckweb;” to “USE whckweb;”
7. Execute the script dbo.tsiewhckweb\_CREATE\_DB.sql to create a new database.
8. In SQL Management Studio, at the top level security node under Local Sql Server 🡪 Security 🡪 Logins, right click the Logins node, and select “New Login”.
9. Under “Select a page 🡪 General Tab”, set the Login name as “whckweb”, select “SQL Server authentication”, and set the password to “Whckweb123”.
10. Unselect “Enforce password policy”, “Enforce password expiration”, and also “User must change password at next login”.
11. Change the “Default database” selection from “master” database to “whckweb” database.
12. Still on the Login – New dialogue box, under “Select a Page” on the left, click on “User Mapping”, and click the checkbox “User mapped to this login” for the “whckweb” database that was created in step G). That click should auto-popolate User “whckweb” to be created, and also “Default Schema” must be left BLANK. The default schema will be filled in automatically when (but not yet!) “OK” is clicked on this Login – New dialogue box.
13. Click “OK” to close the “Login - New” dialogue box.
14. Under Local Database 🡪 Databases 🡪 whckweb 🡪 Security 🡪 Users 🡪 whckweb, right click the user whckweb, and click properties. Under “Database role membership”, please select to add “db\_owner” as primary role membership for whckweb database user.
15. You now have a valid whckweb database user login for the web application <http://tsieauto1/whckweb> to use for its database.
16. Next Steps are inside of Visual Studio that will recreate the ADO.NET Entity Data Model file tsiewhckweb\_SqlDbModel.edmx which directly specifies the SQL connection string to this database just now created using also the database user “whckweb” also just now created on the production server tsievauto1.
17. On your local machine, Open Visual Studio. Our current version that we are running for development is “Microsoft Visual Studio Ultimate 2012 RC”.
18. From source control copied to your local machine, open the solution file “tsiewhckweb.sln” inside of Visual Studio.
19. Open the “Server Explorer” window, and then click the icon or menu item to “add a new connection”.
20. In the “Add Connection” dialogue box, set the following items:
21. Data source left as “Microsoft SQL Server (SqlClient)”
22. Server name as “tsievauto1”
23. Use SQL Server Authentication as User name: “whckweb” minus the quotes, and also the Password: “Whckweb123” minus the quotes.
24. Connect to a database: Select or enter a database name: “whckweb” minus the quotes.
25. Click “Test Connection” to ensure the server firewall is configured properly, and also this authentication as one complete connection string properly works.
26. Click “OK” to add this new SQL connection to Server Explorer.
27. In the Solution Explorer, open the top most level “web.config” file, under the XML data node <configuration><connectionStrings></connectionStrings></configuration>, find and delete \*ALL\* connection string entries that start with the <add></add> node, save the top most solution level web.config file, and close it..
28. Again inside the Solution Explorer, right click the file “tsiewhckweb\_SqlDbModel.edmx”, and click “Delete”, then “OK” to remove it permanently.
29. Right click the “Models” folder, and then point to “Add”, and then “New Item”.
30. Under Visual C# 🡪 Data, select “ADO.NET Entity Data Model”, enter the name: EXACTLY as “tsiewhckweb\_SqlDbModel.edmx” minus the quotation marks, and click the “Add” button to add this again to the Models folder.
31. Keep selected “Generate from database”, and click Next.
32. Under the question prompt “Which data connection should you application use to connect to the database?”, select the entry found there as “tsievauto1.whckweb.dbo”.
33. Click the radio button “No, exclude sensitive data from the connection string”.
34. Under “Save entity connection settings in Web.Config”, set the name as “tsiewhckwebEntities”, and click “Next”.
35. Under “Which database object do you want to include in your model”, click only the Tables check box, BUT THEN exclude the table that says “sysdiagrams”. Do not check Views, and do not check Stored Procedures and Functions.
36. Leave check the three check boxes, and enter the name “tsiewhckweb.Models” for the Model Namespace., and then click “Finish”.
37. After that is closed, you will see the dialogue box “Security Warning”. Please click “OK” to close each of them as the text template files already in the solution explorer are now regenerating all the required DbSet MVC databases for the web application to connect to the database whckweb to use it.
38. Click on the file “tsiewhckwebSqlDbModel.edmx”, then click in the “Properties” Window, click on the “Custom Tool Namespace”, and enter the namespace text “tsiewhckweb.Models”.
39. On the keyboard, press the key combo Ctrl+S to SAVE the tsiewhckweb\_SqlDbModel.edmx file, and regenerate again the DbSet MVC required C# classes code from the 3 text template files also saved there in the “Model” folder.
40. \*IMPORTANT\* - In the solution explorer, click the file “tsiewhckweb\_SqlDbModel.Designer.cs”, and set the “Build Action” of that file there to “Content” to not include that file in the build.
41. In the Solution Explorer, right click the main web application project file found there, and click on “Rebuild Solution” to rebuild the entire web application.
42. After the Rebuild Solution has succeeded, again in the solutions explorer, right click the project file “tsiewhckweb”, and click “Publish…”.
43. Select the entry you see there as “Publish\_to\_tsievauto1(Production)”, and click “Publish”.
44. If rather lengthy steps were done in order, and you see no errors in Visual Studio error window/output window, then the site published OKAY and is ready for testing with the user base!