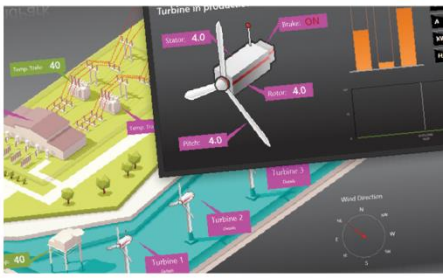
INDUSTRIAL AUTOMATION SOFTWARE
progea

Movicon.next™
Automation Platform.next generation



Movicon NExT

11.0 Webserver

Ver.3.4.268

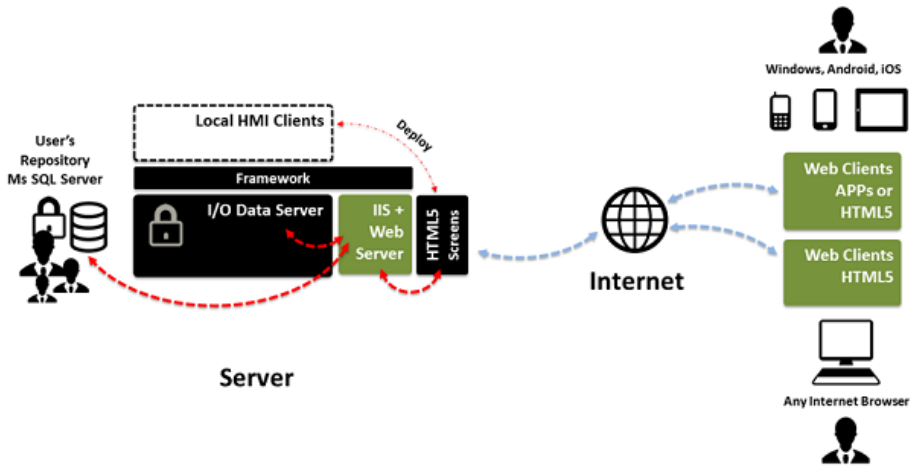
Table of Contents

1. WebServer HTML5

1.1. Introduction

The Platform.NExT Web Client technology is the most modern and innovative solution to use for managing remote access to the supervision system from any remote device or mobile by means of using local and public networks, VPN or internet with the maximum of simplicity and security.
The use of the Platform.NExT Web Client is optional and must be enabled only on the Server's runtime license in order to allow access to the desired number of concurrent HTML5 (Browser) or WPF (Apps) users.
The diagram below illustrates an example of the Platform.NExT web architecture. As you can see, the Data Server uses a Web Server application (IIS for HTML5 or integrated for WPF) and the screens, that have been created, can be published on HTML5 web pages or by means of an App using the relevant Editor command.

In this way users can independently access the Web pages and interact with the system by displaying screens. User login can also be implemented by configuring users and their passwords in the data server.



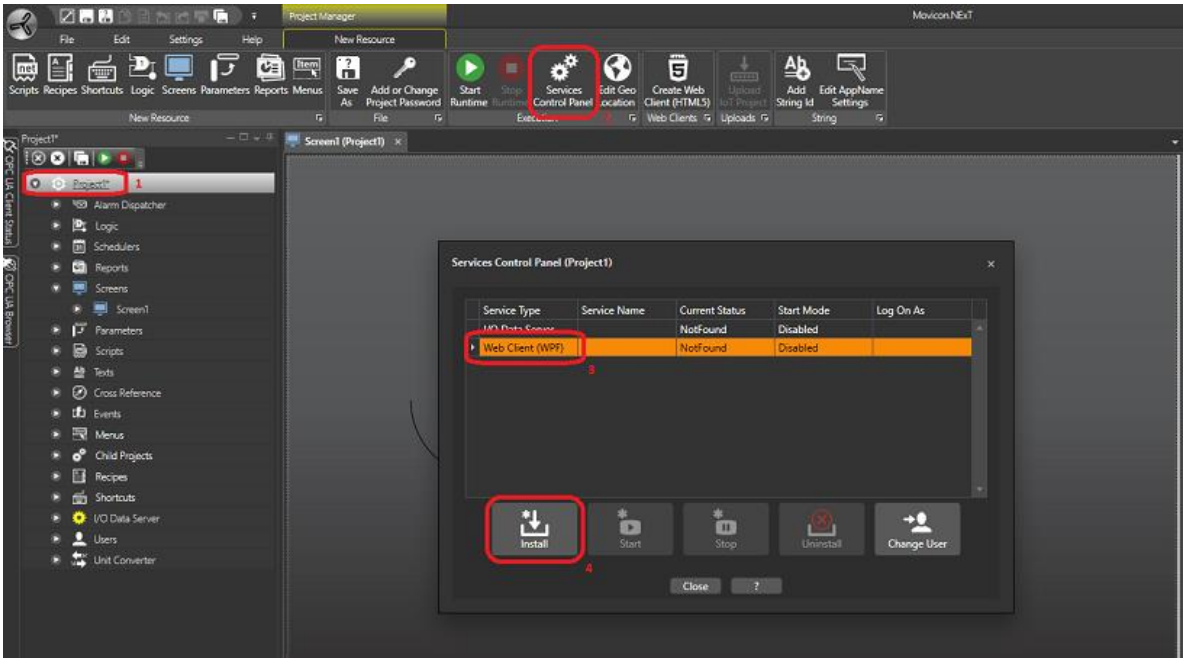
The Platform.NExT technology enables you to use the Platform's Web Server features to deploy the supervision project on the Web and therefore to be accessed by remote Client devices using the HTML5 technology. Thanks to this new standard, Clients can access the server adopting the "Cross Platform" concept whereby the client can access the server using any browser and device with any operating system without having to install extra locally to enable this..

1.2. WPF (App)

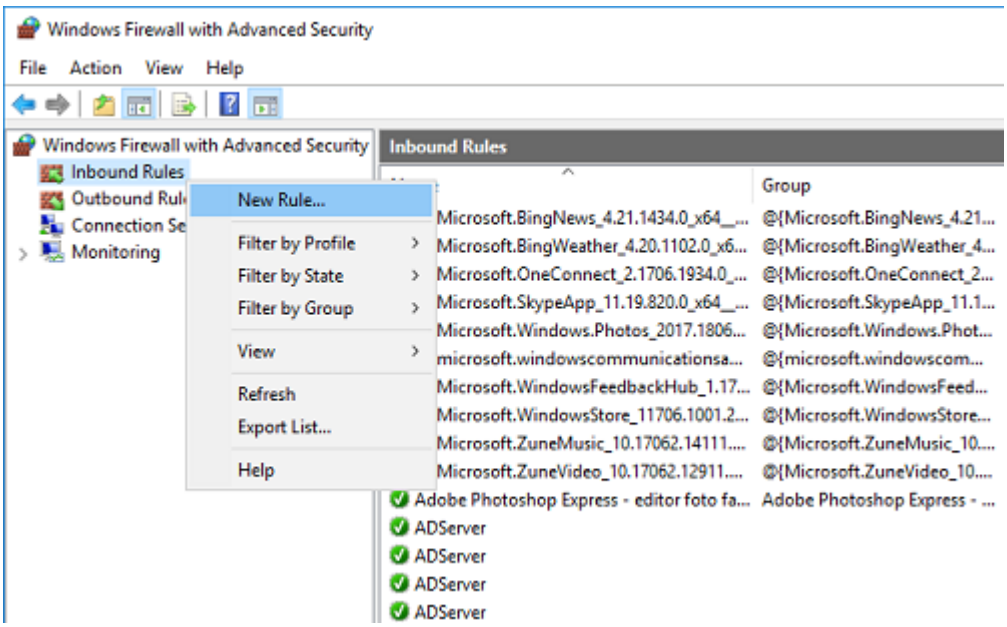
1.2.1. Deploying Projects

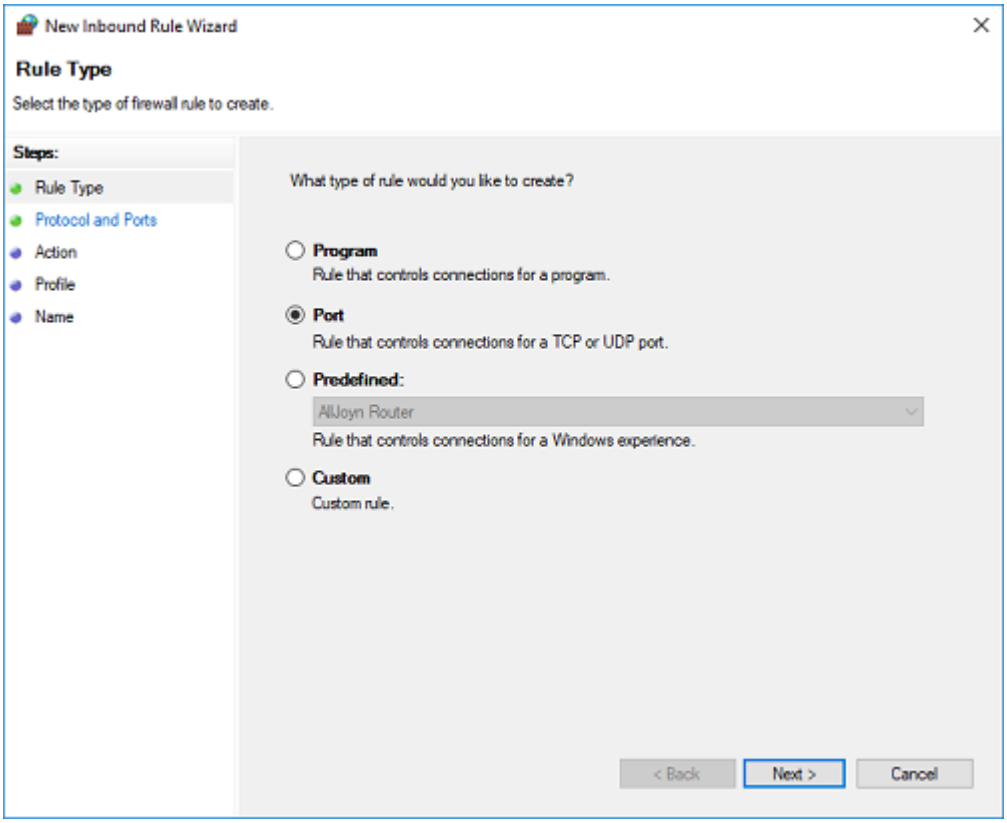
Movicon.NExT offers an app that you can use as a remote client to connect to your application's screens in a completely independent way rather than using Desktop client or HTML5.

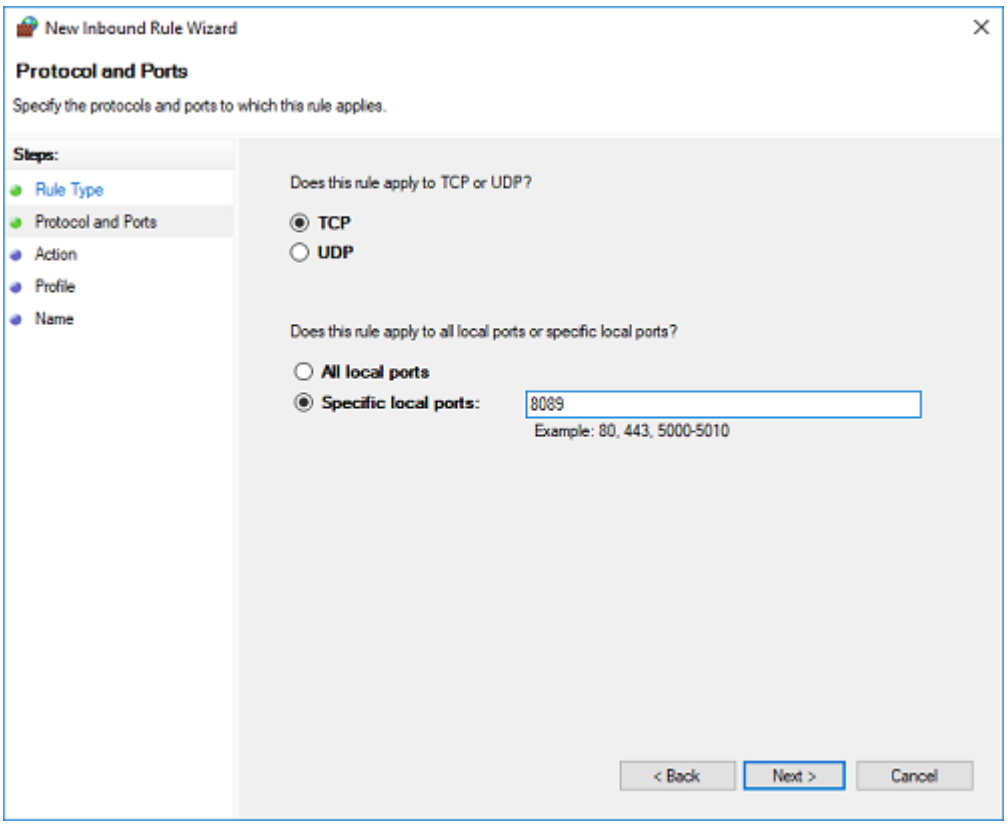
In order to permit the app to connect to the NExT Server that deployed the project, you will need to install and launch the "Web Client (WPF)" service from the "Service Control Panel" as shown in the image below:

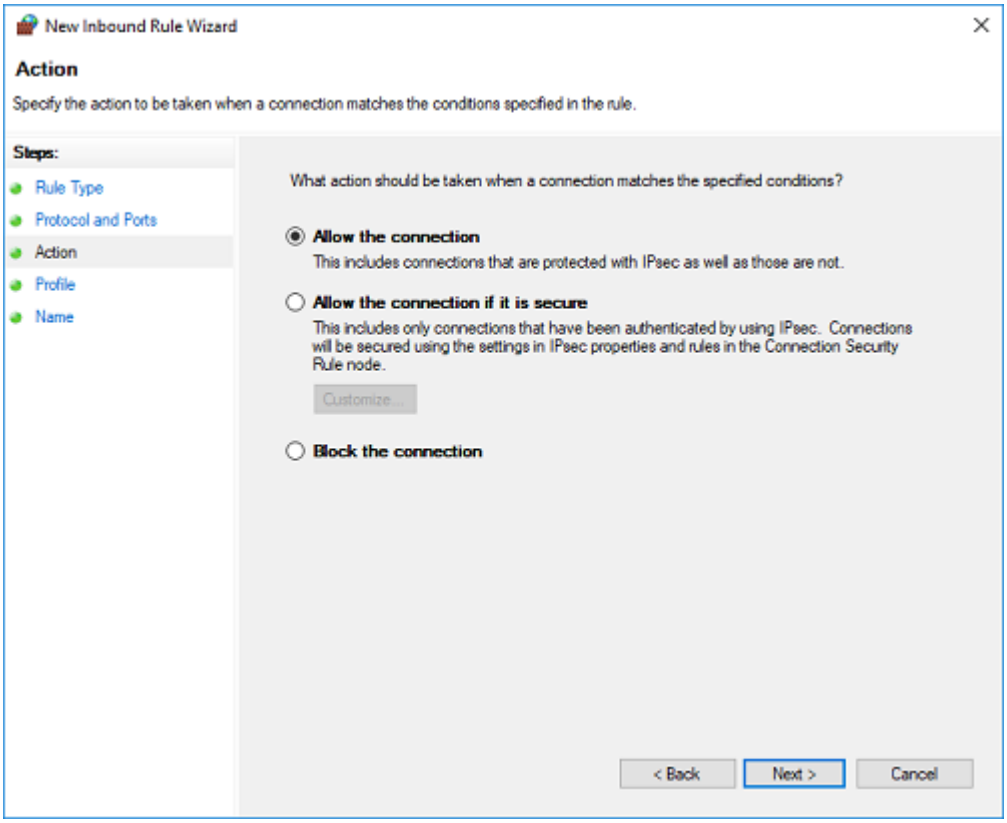


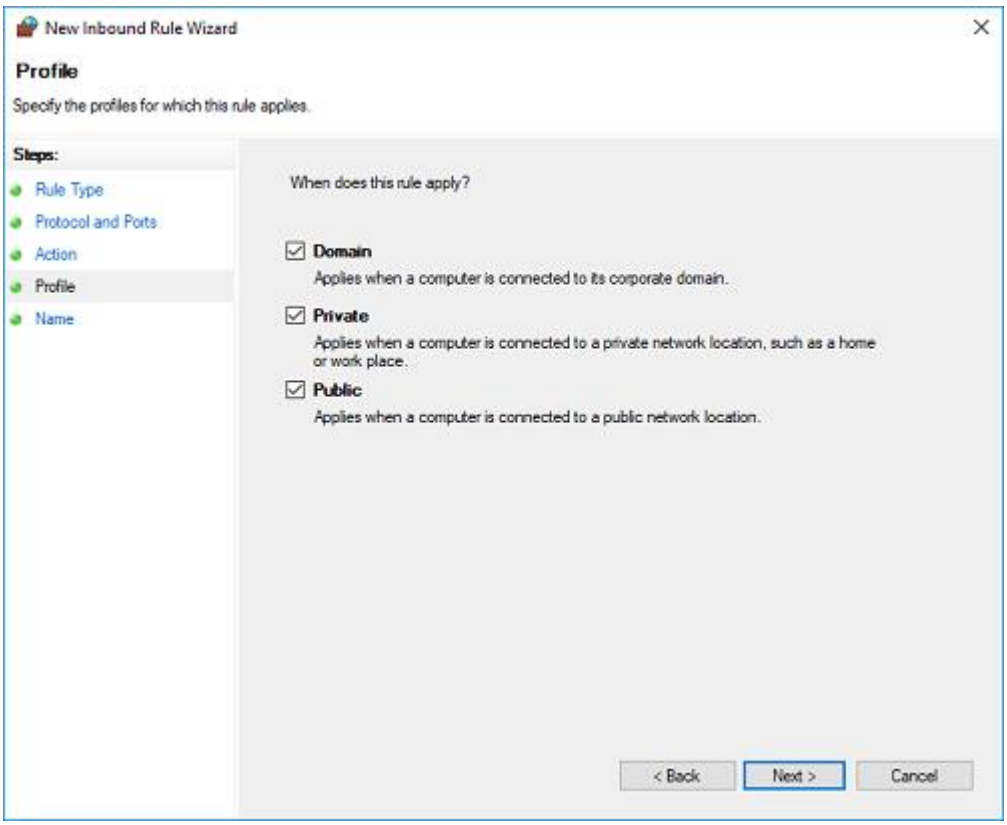
Once the service has been installed an launched, it might be necessary to configure the active Firewall or Antivirus to allow the use of the 8089 port (for default) for the TCP protocol.
The Windows Firewall should be configured as follows:

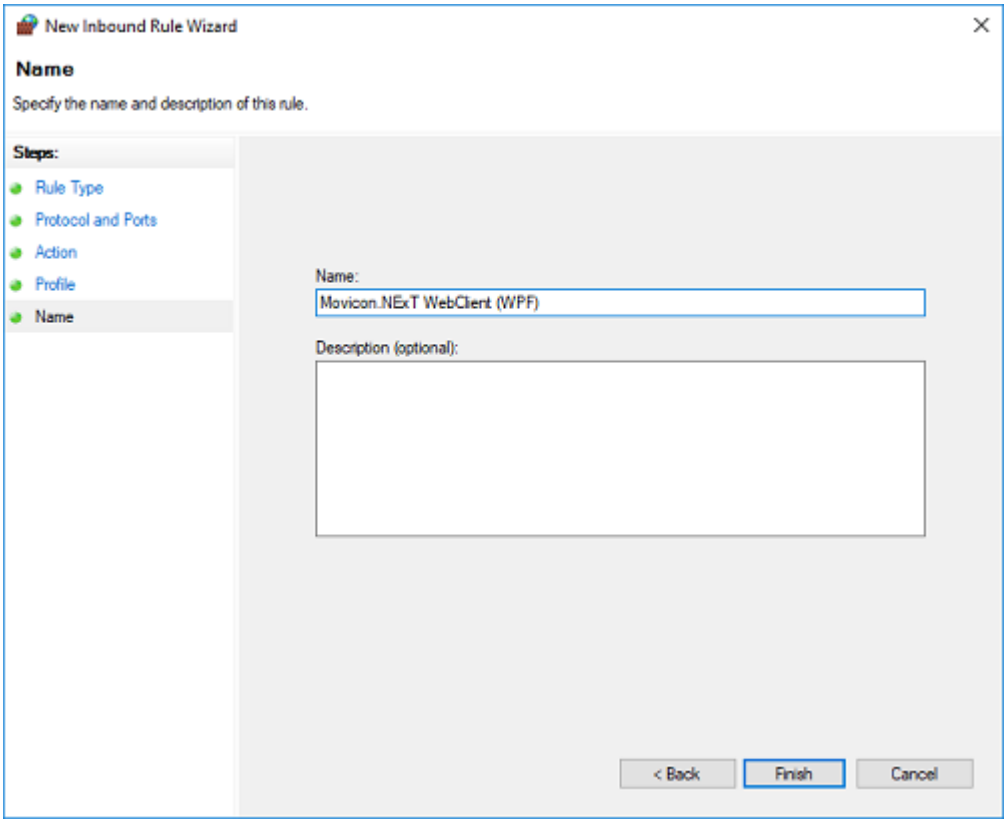

Windows Firewall with Advanced Security
File Action View Help
Windows Firewall with Advanced Security
Inbound Rules
Outbound Rul
Connection Se
Monitoring
New Rule...
Filter by Profile
Filter by State
Filter by Group
View
Refresh
Export List...
Help
Inbound Rules
Group
Microsoft.BingNews_4.21.1434.0_x64_... @{Microsoft.BingNews_4.21...
Microsoft.BingWeather_4.20.1102.0_x6... @{Microsoft.BingWeather_4...
Microsoft.OneConnect_2.1706.1934.0_... @{Microsoft.OneConnect_2...
Microsoft.SkypeApp_11.19.820.0_x64_... @{Microsoft.SkypeApp_11.1...
Microsoft.Windows.Photos_2017.1806... @{Microsoft.Windows.Phot...
microsoft.windowscommunicationsa... @{microsoft.windowscom...
Microsoft.WindowsFeedbackHub_1.17... @{Microsoft.WindowsFeed...
Microsoft.WindowsStore_11706.1001.2... @{Microsoft.WindowsStore...
Microsoft.ZuneMusic_10.17062.14111.... @{Microsoft.ZuneMusic_10....
Microsoft.ZuneVideo_10.17062.12911.... @{Microsoft.ZuneVideo_10....
Adobe Photoshop Express - editor foto fa... Adobe Photoshop Express - ...
ADServer
ADServer
ADServer
ADServer


New Inbound Rule Wizard
Rule Type
Select the type of firewall rule to create.
Steps:
Rule Type
Protocol and Ports
Action
Profile
Name
What type of rule would you like to create?
Program
Rule that controls connections for a program.
Port
Rule that controls connections for a TCP or UDP port.
Predefined:
AllJoyn Router
Rule that controls connections for a Windows experience.
Custom
Custom rule.
< Back
Next >
Cancel


New Inbound Rule Wizard
Protocol and Ports
Specify the protocols and ports to which this rule applies.
Steps:
Rule Type
Protocol and Ports
Action
Profile
Name
Does this rule apply to TCP or UDP?
TCP
UDP
Does this rule apply to all local ports or specific local ports?
All local ports
Specific local ports:
8089
Example: 80, 443, 5000-5010
< Back
Next >
Cancel


New Inbound Rule Wizard
Action
Specify the action to be taken when a connection matches the conditions specified in the rule.
Steps:
Rule Type
Protocol and Ports
Action
Profile
Name
What action should be taken when a connection matches the specified conditions?
Allow the connection
This includes connections that are protected with IPsec as well as those are not.
Allow the connection if it is secure
This includes only connections that have been authenticated by using IPsec. Connections will be secured using the settings in IPsec properties and rules in the Connection Security Rule node.
Customize...
Block the connection
< Back
Next >
Cancel





Attention! To allow th app to connect to the I/O Data Server, the following conditions must be met:

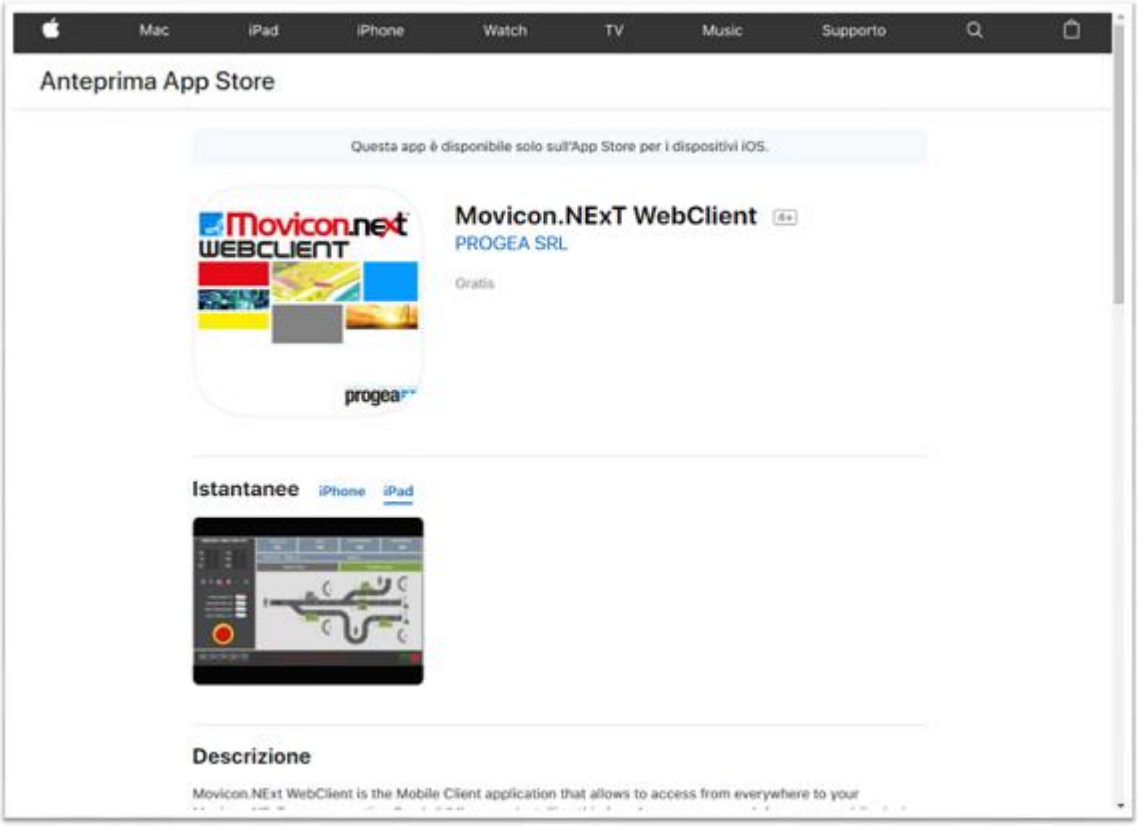
- The I/O Data Server must be installed and started up as Windows service.
- The I/O Data Server must be started up as Administrator (when launching project from editor, open editor as administrator).
- Only set the net.tcp as transport type in the I/O Data Server.

• Add the net.tcp transport type in addition to net.pipe in the I/O Data Server and edit the App Name Settings with the EndpointRenamed parameter equal to net.tcp://localhost:62846/<app_name_of_project>

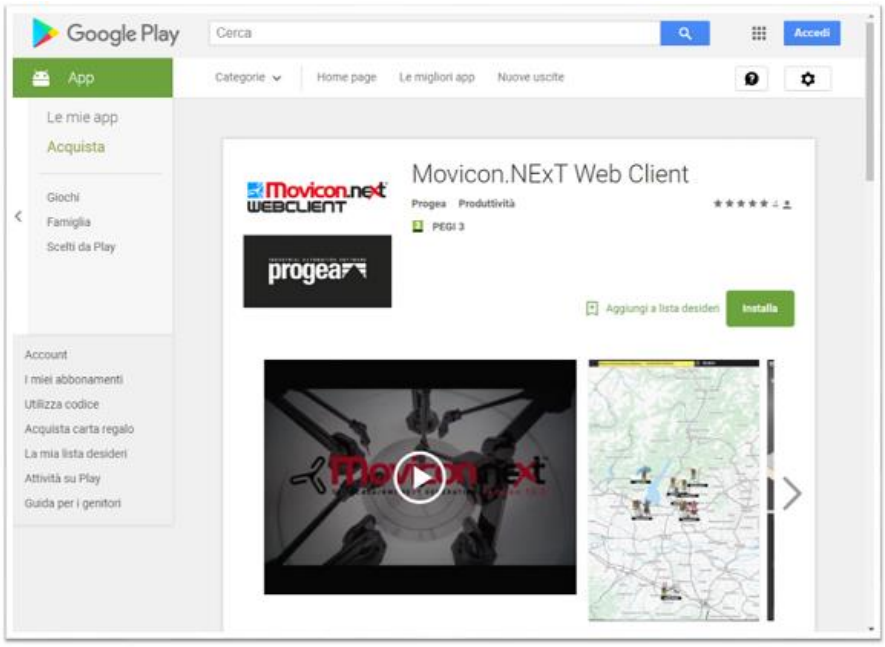
⚠ Attention! When the project involves the use of Users, you will need to enable the user used for starting up the Web Client (WPF) service between the sysadmin of the SQL Server instance.
The user predefined for the Web Client (WPF) service is “LocalSystem” identified in SQL as “NT AUTHORITY\SYSTEM”

The Movicon.NEXT WebClient app is available for free from:

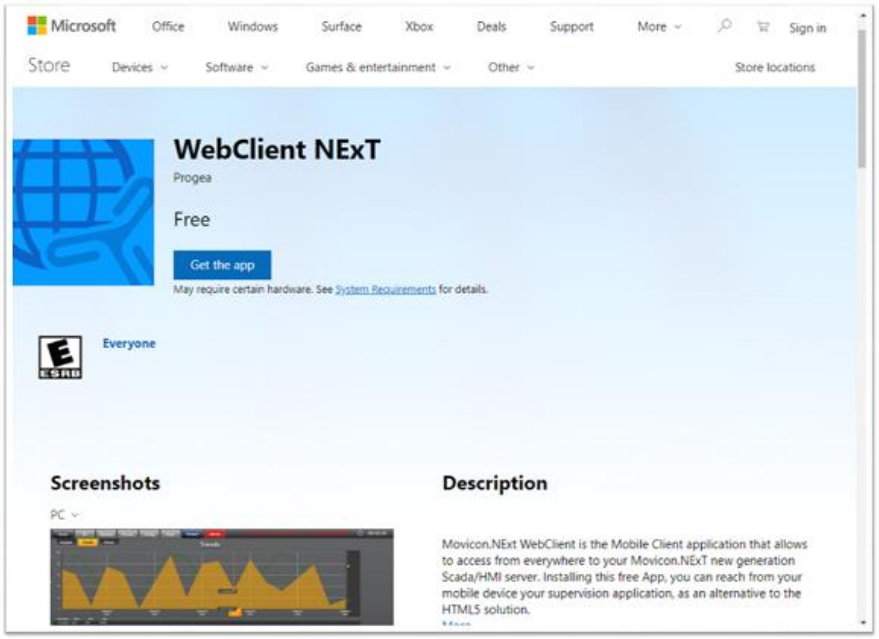
- The App Store for iOS smartphones/tablets



- Google Play for Android smartphones/tablets



- Microsoft Store for Windows smartphones, Desktops and wearable devices (Hololens)



1.2.2. App Limitations

App Limitations

Visualizing pages with the app has some functional limitations that the design engineer should take into consideration when designing their project. They must evaluate whether to use the App or HTML technology based on their specific project needs. Each architecture has its own advantages and disadvantages which must be taken into account accordingly:

App Advantages:

- Service integrated in NExT with needing Microsoft IIS

- The use of specific apps for mobile devices

HTML5 Advantages:

- Use of Microsoft IIS with option to use redundant infrastructure for multiple Web Client service.
- Any Brower can be used.

Attention! If the design engineer wants to make the project fully usable by app, they must take special care that such limits do not prohibit users from using the necessary functions correctly.

WPF Web Client side Limitations (App)

Please refer to the the Web Client WPF (App) section in the topic on "Technical specifications".

1.3. HTML5 (Browser)

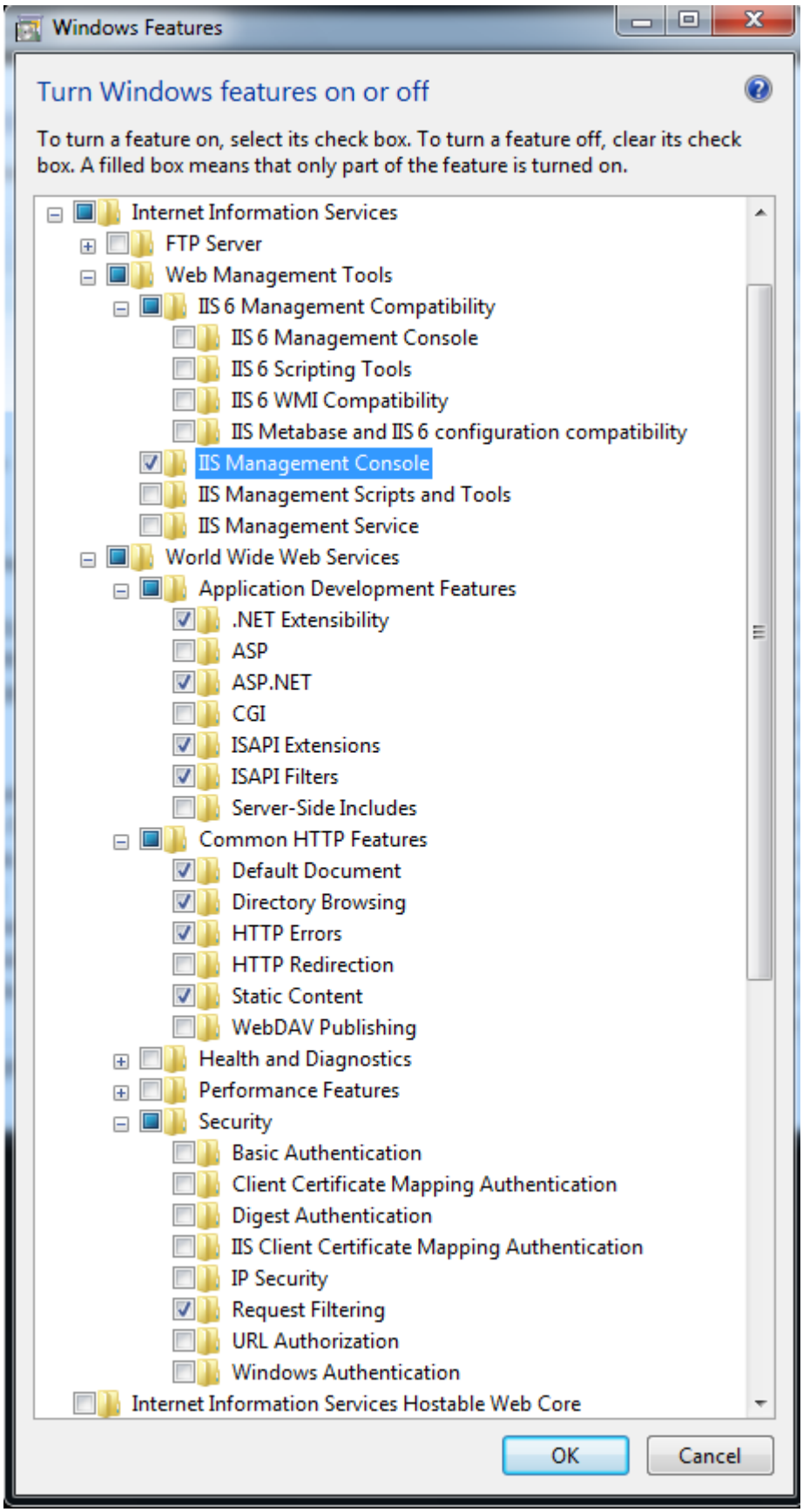
1.3.1. Server Configuration

In order to use the Platform.NExT Web Server technology the following configurations are required:

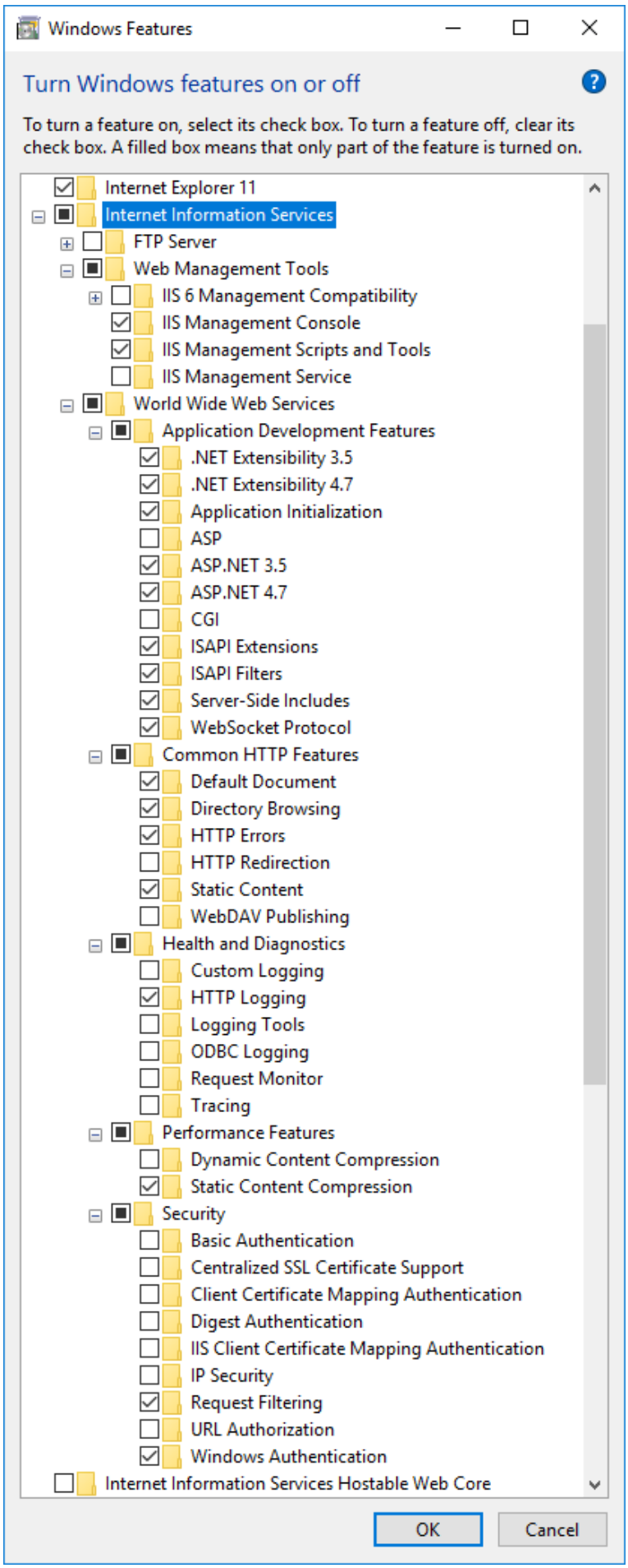
1. Platform.NExT I/O Data Server with Web Server module enabled
2. Windows Server 2008 or 2012 operating system. Windows 8 or Window 7 Desktop version (Workstation) but is limited to 10 connections.
3. The Windows IIS features in the 7.5 version or later (8.0 version is preferable available in Windows Server 2012 or Windows 8)
4. Network connection

In order for the Platform.NExT Web Server module to use the Windows system features you will need to make sure that these features are present and enabled as described below and refer to the -> Control Panel -> Progams and Features > Turn Windows features on or off.

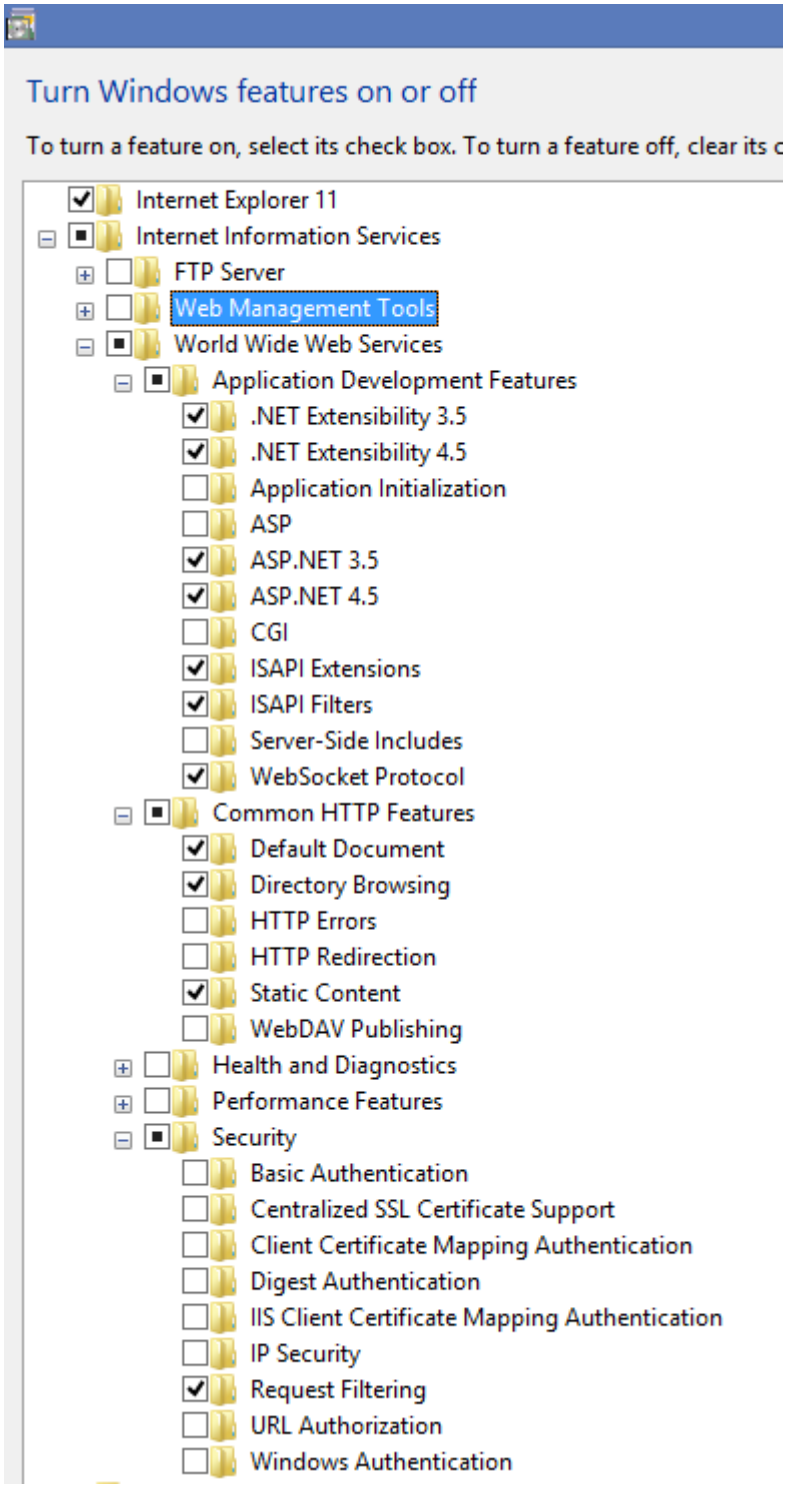
The image below shows the Windows features that must be installed in Windows 7 (IIS 7.5):



The image below shows the Windows features that must be installed in Windows 10 (IIS 10.0):

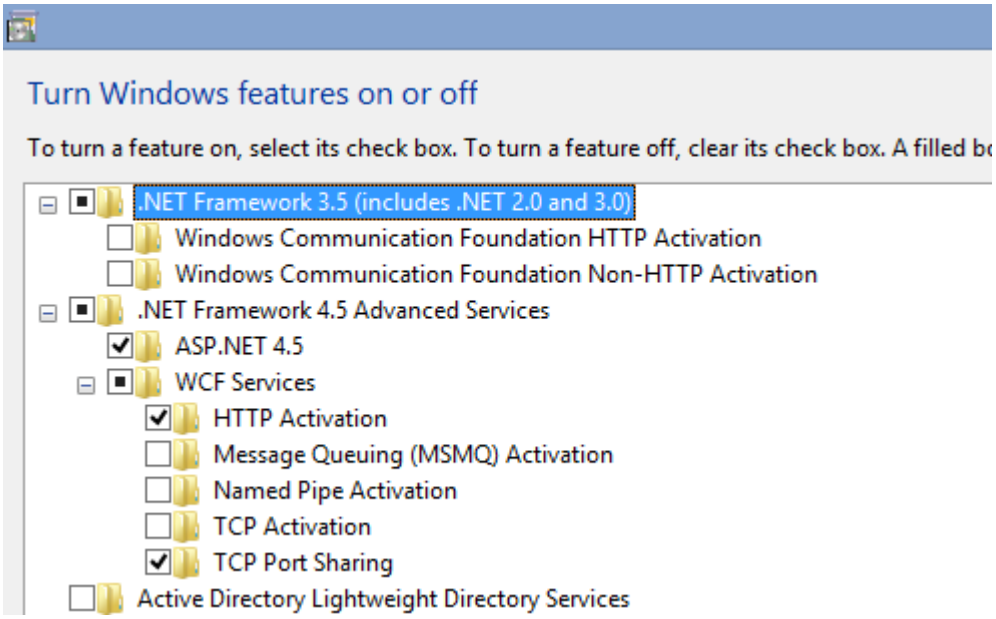

Windows Features
Turn Windows features on or off
To turn a feature on, select its check box. To turn a feature off, clear its check box. A filled box means that only part of the feature is turned on.
Internet Explorer 11
Internet Information Services
FTP Server
Web Management Tools
IIS 6 Management Compatibility
IIS Management Console
IIS Management Scripts and Tools
IIS Management Service
World Wide Web Services
Application Development Features
.NET Extensibility 3.5
.NET Extensibility 4.7
Application Initialization
ASP
ASP.NET 3.5
ASP.NET 4.7
CGI
ISAPI Extensions
ISAPI Filters
Server-Side Includes
WebSocket Protocol
Common HTTP Features
Default Document
Directory Browsing
HTTP Errors
HTTP Redirection
Static Content
WebDAV Publishing
Health and Diagnostics
Custom Logging
HTTP Logging
Logging Tools
ODBC Logging
Request Monitor
Tracing
Performance Features
Dynamic Content Compression
Static Content Compression
Security
Basic Authentication
Centralized SSL Certificate Support
Client Certificate Mapping Authentication
Digest Authentication
IIS Client Certificate Mapping Authentication
IP Security
Request Filtering
URL Authorization
Windows Authentication
Internet Information Services Hostable Web Core
OK
Cancel

The image below shows the Windows features that must be installed in Windows 8 (IIS 8.0):



Other required features:

In addition to these options you will also need to explicitly select those options relating to the WCF services (not available in win7).



1.3.2. Client Limitations

The visualization of graphical pages on the HTML5 Web Client side is limited in how it functions due to its intrinsic Web and Cross Platform features and which must be taking into consideration by the user when designing their project.
Based on the project's needs on the Client side the user should assess whether to use the Web Client or Desktop Client technology or even the Terminal Server technology (Remote Desktop).
Some architectures have both advantages and disadvantages that must be assessed in function with project needs.

Attention! It is important that the project designer, who is responsible for making the project completely usable by Web Clients, take into full consideration existing web architecture restrictions in order to provide the user with the right functionalities they need.

To resolve eventual references to the Movicon ToolBox assembly in project scripts correctly, you will need to copy the entire ToolBox folder, contained in the Movicon installation folder (c:\Program Files\Progea\Movicon 3.x), to the C:\Windows\System32\inetsrv" folder.

Web Client side limitations

Please refer to the Topic on "Technical Specifications" in the Web Client section:

Antivirus

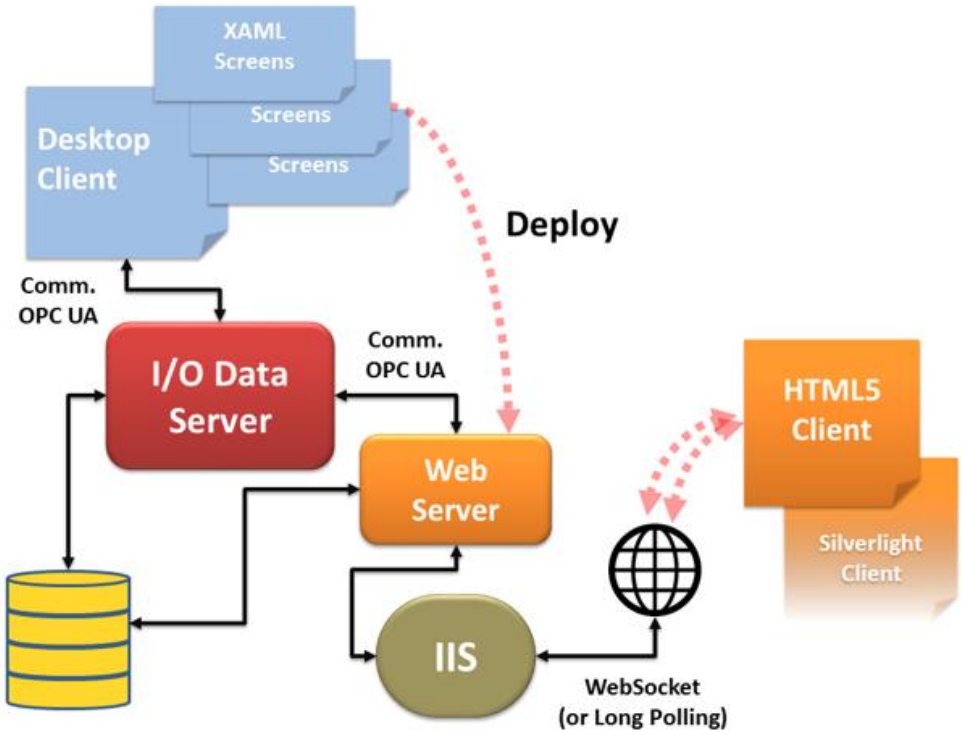
Using an antivirus that has online data traffic analysis functions might slow down the Web Client performances. Therefore it would be best to take into consideration that communication performances may be effected when the antivirus attempts to analyse or filter all the data packets transmitted by online protocols. If there are any problems it is advised to disable the antivirus's online data analysis while using the client (for example, you may have to disable the "online shield" function when using the AVG antivirus).

Page Refresh

The browser's page "refresh" command (F5), returns to the initial "Tile Page" and restarts the Web Client application.

1.3.3. Web Server Architecture

The system architecture has been designed with a Web Server module for the Platform.NExT platform that manages connections to the I/O Data Server and HTML5 client pages using the OPC UA communication technology. In this way the Web Server's task is to ensure real-time communication with the data server by enabling client access using Windows IIS Web Server.



Therefore as with the Movicon.NExT graphical interface for desktop clients, the Web Server acts as an OPC UA client towards the data Server.

This Web architecture is totally independent. Therefore it is possible to create projects that have both a local Client HMI interface and a HTML5 Web Client interface. Otherwise a "blind" server can be implemented to publish graphical pages only for remote Web Clients.

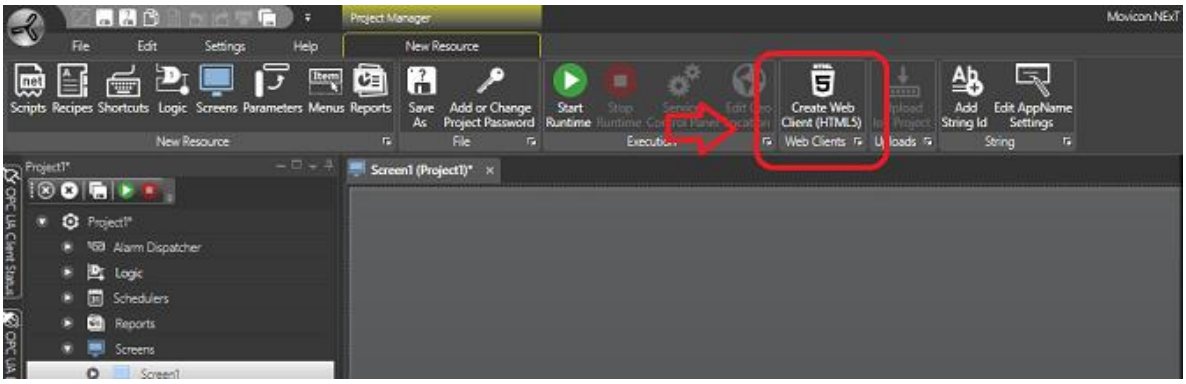
Access to the Supervisor is performed on the Client side using a common internet browser (Internet Explorer, Chrome, Firefox, Safari, etc.), with different devices such as PC, Tablet and Smartphone. the HTML5 Web technology has the advantage of being Cross Platform-based because it is practically available in all the most popular operating systems (Windows, Windows Phone, iOS, Android, Linux, etc.).

Thanks to the HTML5 technology, clients are no longer passive and get to manage bidirectional communications, user protection, database access and report presentations, within a Web context without needing any installation or configuration on remote client side.

Even though there are many advantages of having distributed web technology, there are also some limits to its use on the HTML5 Client side as described in this guide. The user should consider these limits when project engineering by evaluating what of the two architectures, Desktop Client or Web Client , is most appropriate to use according to their needs.

1.3.4. Project Deployment

After having created your project in the Movicon.NExT workspace as well as the Tags and screens as needed, you can choose whether to deploy your application on the Web according to the HTML5 Web Client architecture described previously.
To deploy your project you can use the **'Create Web Client (HTML)'** command from the project's contextual ribbon in the Platform.NExT workspace in development mode or use the same command by right clicking on the root of the project's tree structure.



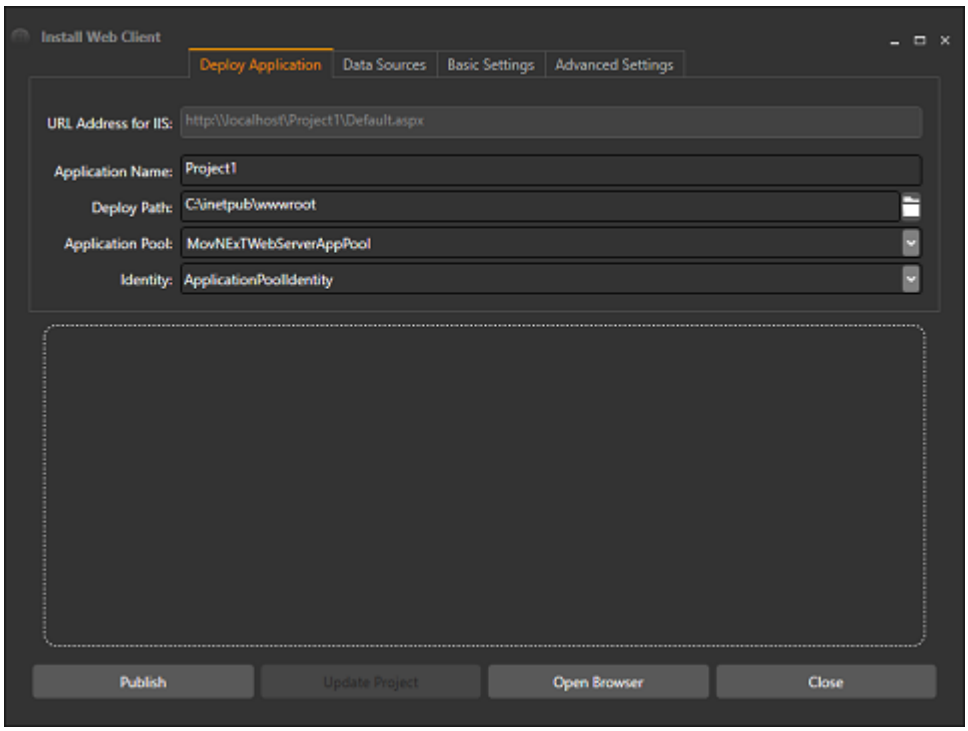
If you are using custom WPF UserControls in screens, in order to use these controls on the WebClient side you will need to copy the relating ".dll" files within this path:

"C:\Program Files\Progea\Movicon.NExT 3.1\UFWebClient.HTML5\bin\Toolbox"

before executing the 'Create Web Client (HTML5) command:

Deploy Application window

The use of the command to deploy projects on the Web, determines the creation of HTML5 pages and the configuration of the connections relating to the platform's Web Server module.
As a consequence, a data settings window will display, as shown below, to allow you to select, modify or configure the deploy parameters.
The Deploy window permits you to define the following deploy parameters:



Url address for IIS

The Deploy Web address of the server to which the client must point to. Specifies the URL address which must be digited in the Client browser to display the application's "Main Web Page".

Application Name

Name of the virtual directory to create for the Web Server. It also allows you to replace the name of the deploy URL folder if need be.

Deployment path

The deploy physical path. Indicates the folder in which the deploy web files are available from.

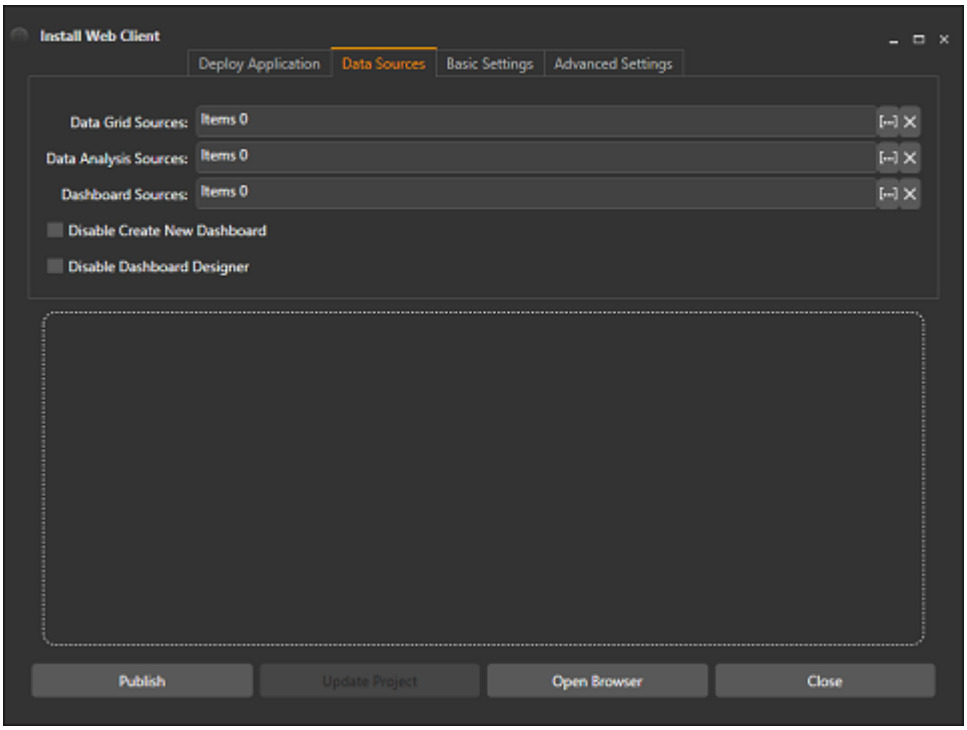
Application Pool

Name of pool with which the web application will be started up. The pool defines the user context within which the Web application is executed and which is necessary, for example, for any eventual access to the SQL Server database managed by the Server project.

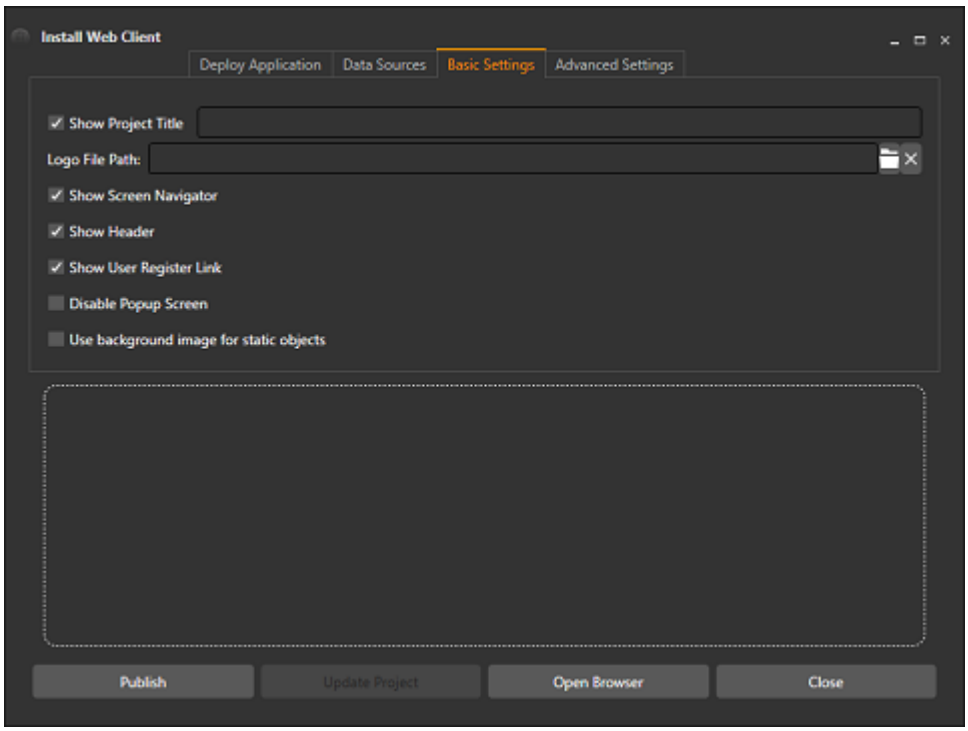
Identity

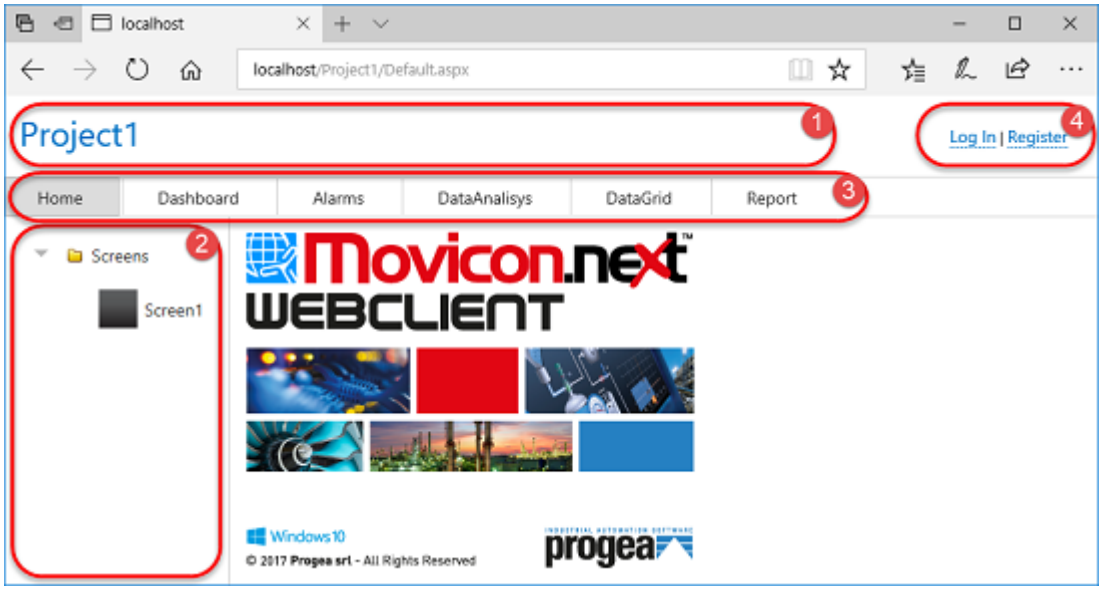
User identity used for the pool. User identity is needed when wishing to access those services outside the project on the server and that request it, such as obtaining access to data on a SQL Server Database. Therefore, if the Web application requests access to a database, it will need to be executed in a pool context with a user idenity which is known to the database within which it has been defined.
Generally, Identity names created for default are proposed and therefore already exist in the Database as Access Authorization Identity. Otherwise, in order to access the contents of a database over the web, you will need to specify an Identity that has been defined in that database on the Server. To verify and/or enter user identities in a SQL

Server database, you will need to run the SQL Server Management Studio.
Proposed users are those which are normally standard and default users:
"LocalSystem","LocalService","ApplicationPoolIdentity","NetworkService".
Otherwise you can indicate a "SpecificUser" (user set as pleased and already defined as one of the users granted access to the DB).



Data Grid Sources	This permits you to set the connection string to the DB which the indicated object will use, in this case the Data Grid, extracting data.
Data Analysis Sources	This permits you to set the connection string to the DB which use the indicated object will use, in this case the Data Analysis, for extracting data.
Dashboard Sources	This permits you to set the connection string to the DB which the object will use, in this case the DashBoard, for extracting data.
Disable Create New Dashboard	Disables the possibility to create new Dashboards.
Disable Dashboard Designer	Disables the Dashboard editor.





Show Project Title

Shows the application name in the web page (see point 1 in screenshot above)

Logo File Path

This property allows you to specify the path of an image to use as a logo in the startup page in Tile Page mode instead of the default Movicon logo.
The Logo path must be a path relating to the application. For IIS security reasons no other folder can be accessed unless explicitly configured to do so. Therefore, to simplify things, the image file must be copied manually to the application folder or one of its sub folders. For example, when deploying the 'TestProject', it will be downloaded in the "wwwroot" with the following path:

C:\inetpub\wwwroot\TestProject

At this point, if we insert the "CustomLogo.png" image in the application's 'Images' sub folder:

C:\inetpub\wwwroot\TestProject\Images\CustomLogo.png

The relative path will be inserted in the Logo File Path property:

Images\CustomLogo.png

When deploying the project the folder will be recreated. Therefore, it will be necessary to c the Logo image file in the path proposed afte deploying project.

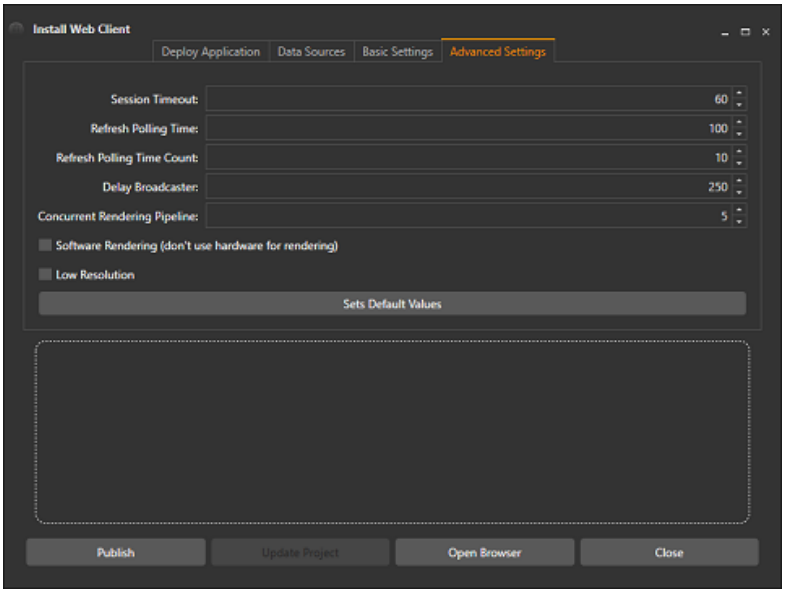
Show Screen Navigator — Shows the screen navigator side panel in the web pages (see point 2 in above screenshot)

Show Header — Shows the toolbar in the web pages (see point 3 in above screenshot)

Show user register link — Shows the "Log In|Register" link in the web pages (see point 4 in above screenshot)

Disable Pop-up Screen — Disables screens opened as pop-ups.

Use Background image for static objects — Activating this option will set all static objects, that are not dynamic, as background image. All the other objects that are dynamic will be handled as usual. In case where you wish to extract a static object from the background image, you will need to activate the 'Force as Dynamic on WebClient' property.



Session Timeout	Inactivity time in seconds after which client session will be considered as expired freeing a WebClient license.
Refresh Polling Time	Refresh time of dynamic objects on screen in milliseconds. When the "Use background image for static objects" property is activated, only the dynamic objects will be refreshed. Otherwise the whole screen will be refreshed.
Refresh Polling Time Count	This represents the number of dynamic objects to be controlled when the time set in the 'Refresh Polling Time' property expires. At each 'Refresh Polling Time Count', the web server application will control and notify connected web clients whether each dynamic object has changed position or image or not.
Delay Broadcaster	This is used to manage broadcasts from the server to the client. Objects that change status are listed and notified to clients after this delay time expires.
Concurrent rendering pipeline	Represents the maximum number of concurrent graphics updates admitted in the web server application for dynamic objects on screens. This value represents a maximum value independently from the number of web clients connects.
Software Rendering	Disables the hardware acceleration to render graphics.
Low Resolution	Uses low resolution images.

The window also has the following command buttons:

Publish	This button is used to publish the HTLM5 web applications.
Update project	This button is used to update the published web application when changes are made to the project files.
Open Browser	Test Button that excutes the opening of the published HTML5 web project using the browser predefined in the operating system. The test may require that the data server be started up in runtime.
Close	Exit button.

Attention! In order to allow the app to connect to the I/O Data Server, at least one of the following conditions must be met:
• The I/O Data Server is installed and started up as Windows Service.
• The I/O Data Server is started up as Administrator (if project is launched from editor, open the editor as administrator).
• Only the net.tcp transport is defined in the I/O Data Server.
• Add the net.tcp transport in addition to the net.pipe in the I/O Data Server and edit the App Name setting with the EndpointRenamed parameter equal to net.tcp://localhost:62846/<app_name_di_progetto>

1.3.5. Web Server Mode

The Web Client browser can also be started up by loading a specific screen page on the server direclty instead of having to select it from the Main page.
Therefore the "ScreenName" must be known within the "Folder name" with the following URL address syntax to be specified when starting up the browser:

http://localhost/ProjectName/Screen.aspx?url=Screen/Foldername/ScreenName.xaml

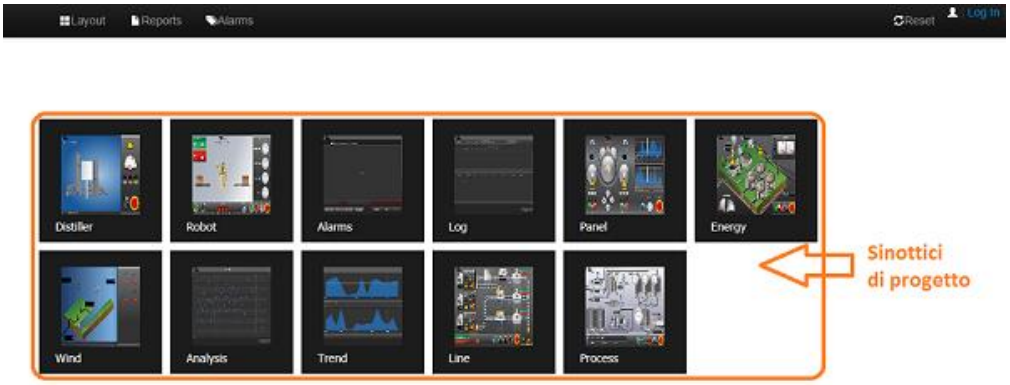
Where the names can be substituted as the following example shows:

http://localhost/DemoMoviconNExT/Screen.aspx?url=Screen/Energy/Energy.xaml

The Web Client page supports different modes: "Tile page", "Main" and "GeoPage",.

Tile Page mode

The list of screens that have been defined in the project will be displayed as tiles on a "Tile page".



There is a menu at the top of the page which is used to access the reports that have been defined in the project, monitor the alarms and to perform Login.

Layout

This link returns you to the intial page displaying the list of screens.

Report

The reports defined in the project are also supported on the HTML Web Client. You will find a "Report" card on the web page displaying all the reports defined in the project

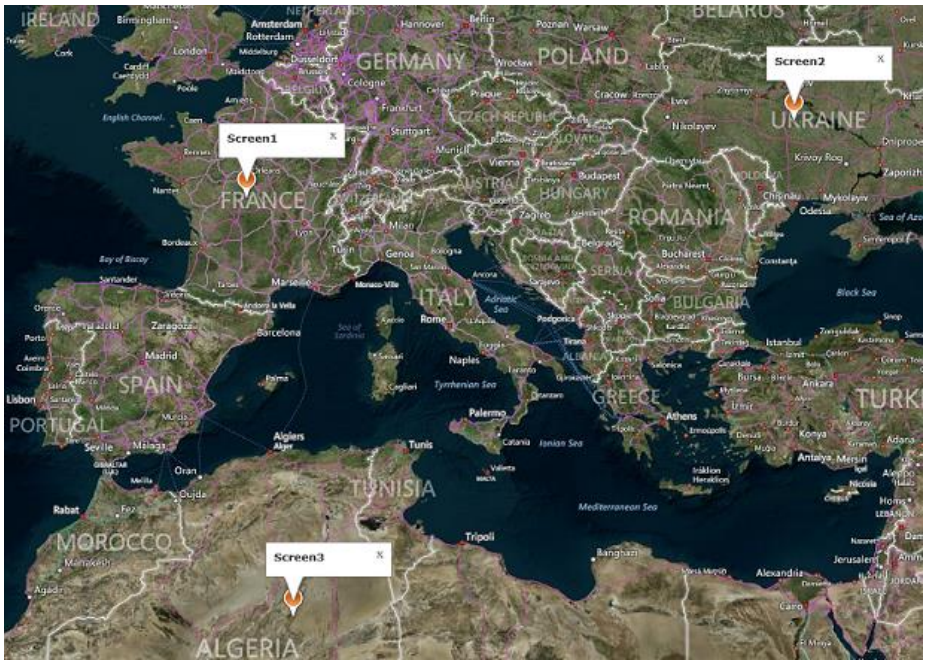
and which can also be displayed on the web side. In addition to the "Alarm Statitics" license option the "AlarmReport.OrderByOccurence","AlarmReport.OrderByDuration" and "AlarmReport.OrderByDateTime" Alarm statistics Reports will also be shown.

Alarms

There is an "Alarms" tab on the web page which can be used to display a page with the Alarm Window.

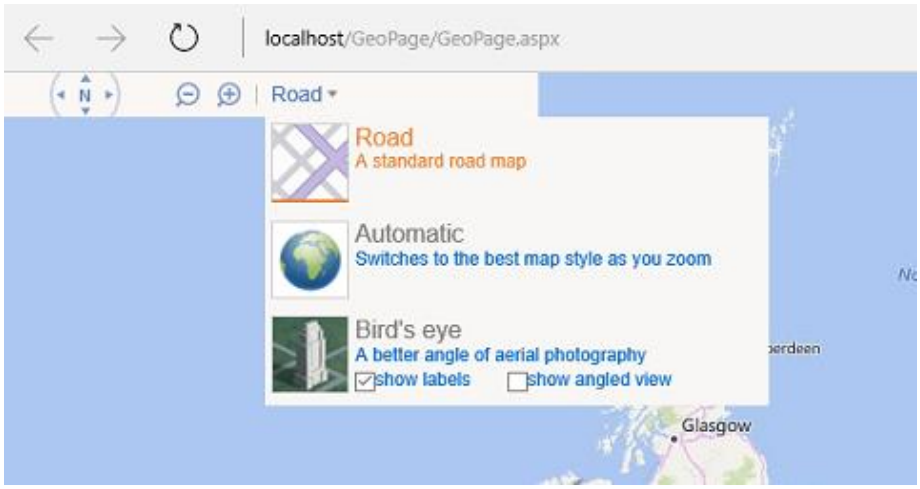
Geo Page Mode

The "GeoPage" startup mode opens up with a map showing a the screen positions. Please see the topic relating to "Geo Page" for further details.



The symbol relating to the screen has a different appearance to the one used on the desktop. When this symbol is clicked on, the screen will open. The name of the screen is shown at the side of the symbol. This name can be hidden the the "x" and restored by refreshing the page. It is also possible to zoom in with a double mouse click or zoom in/out using the mouse wheel.

At the top to the left you will find buttons for managing the zoom, moving geographically and for changing the display mode: road map, geographic map, automatic management, area map.

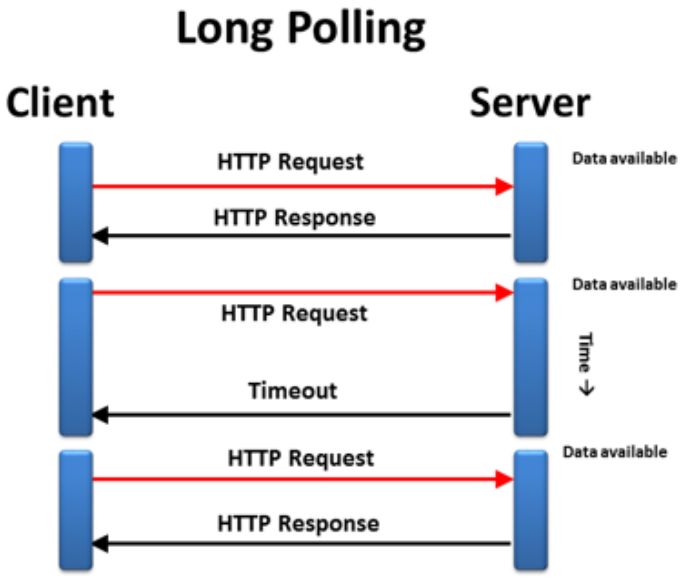


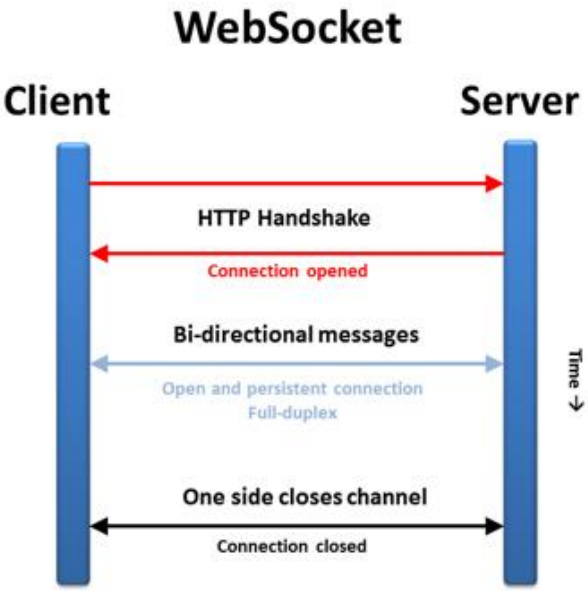
Main Page Mode

When using the Main mode, the project's Main page will load or the first found in alphabetical order.

1.3.6. Server-Client Communication

The Server system and Web Clients communicate with each other in the most natural way that is automatic and transparent to the user. This paragraph will describe the mechanisms involved to inform the user of the technological possibilities open to them depending on the operating system being used.





Communication between web and "Web Server HTML5" pages is performed using different technologies according to the IIS version type.
The IIS 7.5 version component is available in Windows 7, while the 8.0 version is available for Win 8. Communication between web and web server pages is performed using the a WebService with HTTP request.

Long Polling

If the IIS 7.5 is present, the 'Long Polling' communication type is supported. The page sends a request to the Web Server which responds without needing the client to make further requests. The "Long Polling" involves the continuous use of AJAX calls kept open until the data has been received or timeout has expired.

WebSocket

In cases where the IIS 8.0 is available, the WebSocket calls are used as they are better performing than LongPolling. The WebSocket communication is "FullDuplex" and allows two interlocutors to 'speak' and 'listen' at the same time. This overcomes the limited use of the HTTP protocol where calls are sent exclusively: first the request and then the response according to the Half-Duplex mode.
The WebSockets offer a bidirectional channel, both from client to server and viceversa and they are full- duplex to permit simultaneous input and output. Only one layer of http communication is used for handshaking.
In order to be supported on the Server Side, the Websockets require IIS 8.0 offered by Win8 as an optional IIS component.
The table below shows which Browser versions support WebSockets:

	IE	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Blackberry Browser	Opera Mobile	Chrome for Android	Firefox for Android	IE Mobile
29 versions back			4.0										
28 versions back		2.0	5.0										
27 versions back		3.0	6.0										
26 versions back		3.5	7.0										
25 versions back		3.6	8.0										
24 versions back		4.0	9.0										
23 versions back		5.0	10.0										
22 versions back		6.0 Moz	11.0										
21 versions back		7.0 Moz	12.0										
20 versions back		8.0 Moz	13.0										
19 versions back		9.0 Moz	14.0										
18 versions back		10.0 Moz	15.0										
17 versions back		11.0	16.0										
16 versions back		12.0	17.0										
15 versions back		13.0	18.0		9.0								
14 versions back		14.0	19.0		9.5-9.6								
13 versions back		15.0	20.0		10.0-10.1								
12 versions back		16.0	21.0		10.5								
11 versions back		17.0	22.0		10.6								
10 versions back		18.0	23.0		11.0								
9 versions back		19.0	24.0		11.1								
8 versions back		20.0	25.0		11.5								
7 versions back		21.0	26.0	3.1	11.6			2.1					
6 versions back	5.5	22.0	27.0	3.2	12.0			2.2		10.0			
5 versions back	6.0	23.0	28.0	4.0	12.1	3.2		2.3		11.0			
4 versions back	7.0	24.0	29.0	5.0	15.0	4.0-4.1		3.0		11.1			
3 versions back	8.0	25.0	30.0	5.1	16.0	4.2-4.3		4.0		11.5			
2 versions back	9.0	26.0	31.0	6.0	17.0	5.0-5.1		4.1		12.0			
Previous version	10.0	27.0	32.0	6.1	18.0	6.0-6.1		4.2-4.3	7.0	12.1			
Current	11.0	28.0	33.0	7.0	19.0	7.0	5.0-7.0	4.4	10.0	16.0	33.0	26.0	10.0
Near future		29.0	34.0		20.0								
Farther future		30.0	35.0		21.0								
3 versions ahead		31.0	36.0										

Pink = Not supported
Dark green= Partially supported
Light green = supported

1.3.7. Displaying Data in HTML5 Client

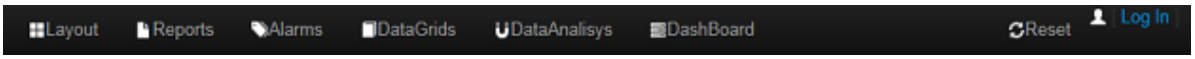
After having configured the server and deployed the project's graphical pages with the Deploy command you will be able to access and display the server project's data using the desired Web Client device such as:

- PC Desktop or Laptop Browser
- Tablet
- Smartphone

The device will act as a "**Thin Web Client**" which will be authenticated and connected to the server when user points to the server's web address (URL) defined when deploying the pages as previously described.
When pointing to the server address, the startup page will be displayed on the HTML5 Web Client terminal.

Main Page command menu

The Web Client's Main Page has a menu on the top border which is used for accessing operating commands for some of the server project's features if provided.
The functions are described below:



Layout

This command calls the Web Client's main page. Therefore invokes the return to the main page.

Report

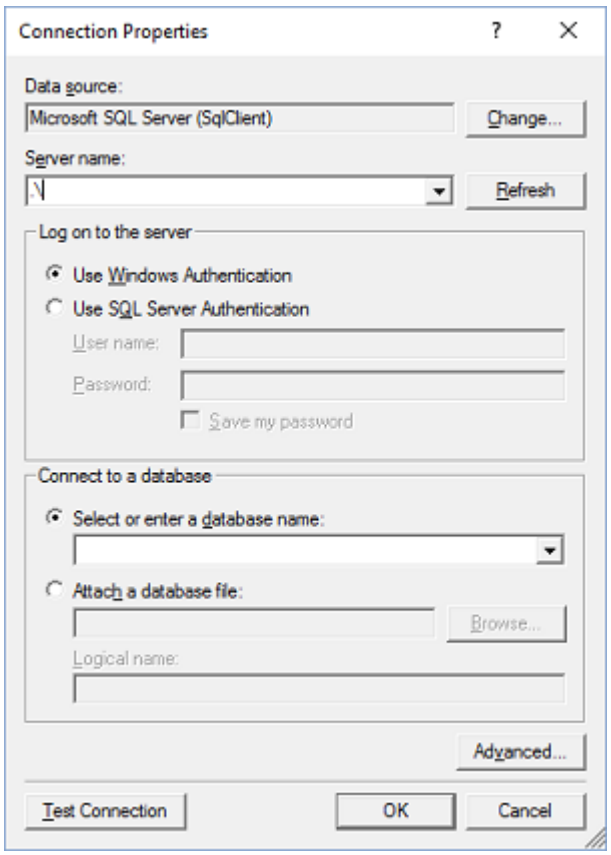
This command is used for viewing Data Report pages on the Client Web side if the Server project has been configured for managing them. This function therefore makes it possible to obtain reports via the web by accessing data recorded on the server to view its contents.

Alarms

This command calls the project's Alarm page and therefore to view active alarms and to perform operations such as sort by, acknowledge and reset also on the Web Client side.

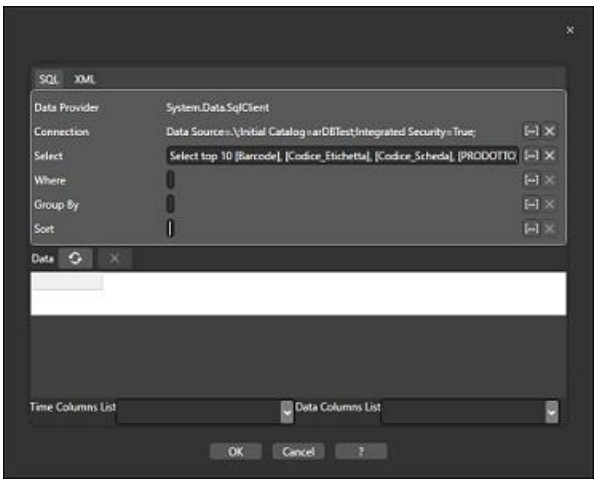
Data Grid

The Data Grid object is particularly useful for representing data in your projects in Grid format. When selecting the Data Grid object from the deploy window while using Microsoft SQL as a provider, for example, a dialog window will show allowing you to define the name of the Server and the Database to be used in order to create a connection string.

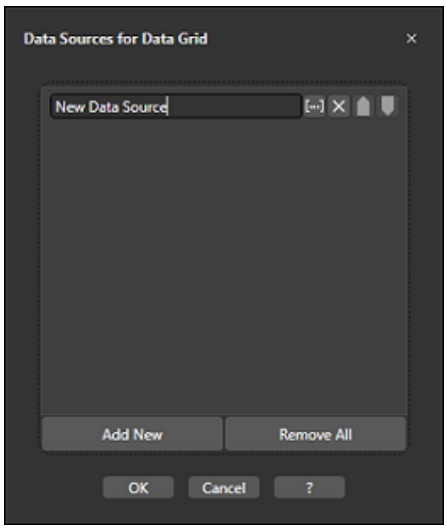


after having defined the your chosen provider you will be asked to enter the name of the server

At this point, insert the query for extracting data from the data source (img.2) in the 'Select' field. This list of data sources used in the project (img.3) will display in the previously opened dialog window.

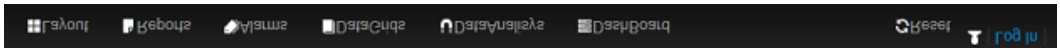


img.2

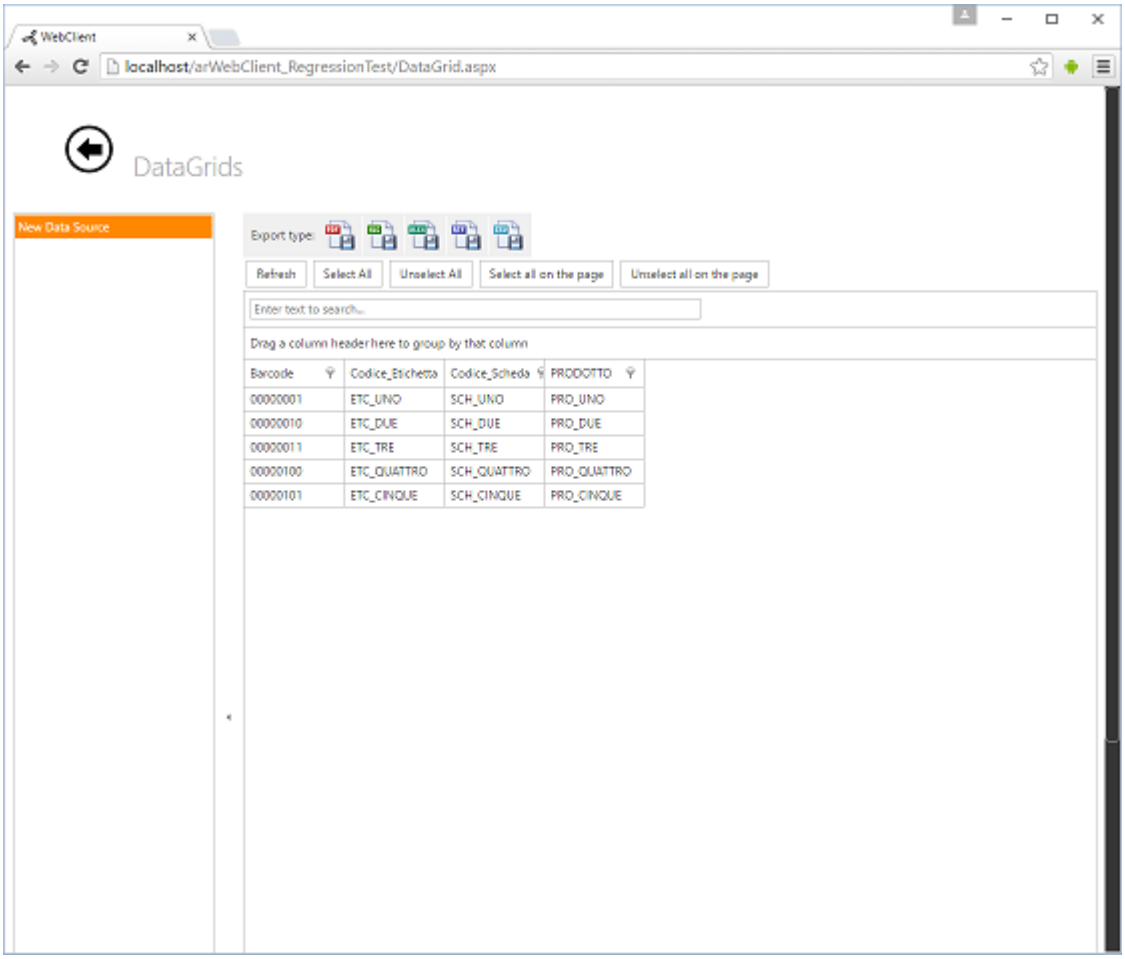


img.3

At this point when you go to publish your project on the WebClient you should see the connections dedicated to the specific objects,which also includes the Data Grid, on the bar at the top.



When you select the Data Grid Resource, all the data sources used until this moment will display. When selecting the data source desired, the data from the Database will be retrieved according to the query that you entered previously.

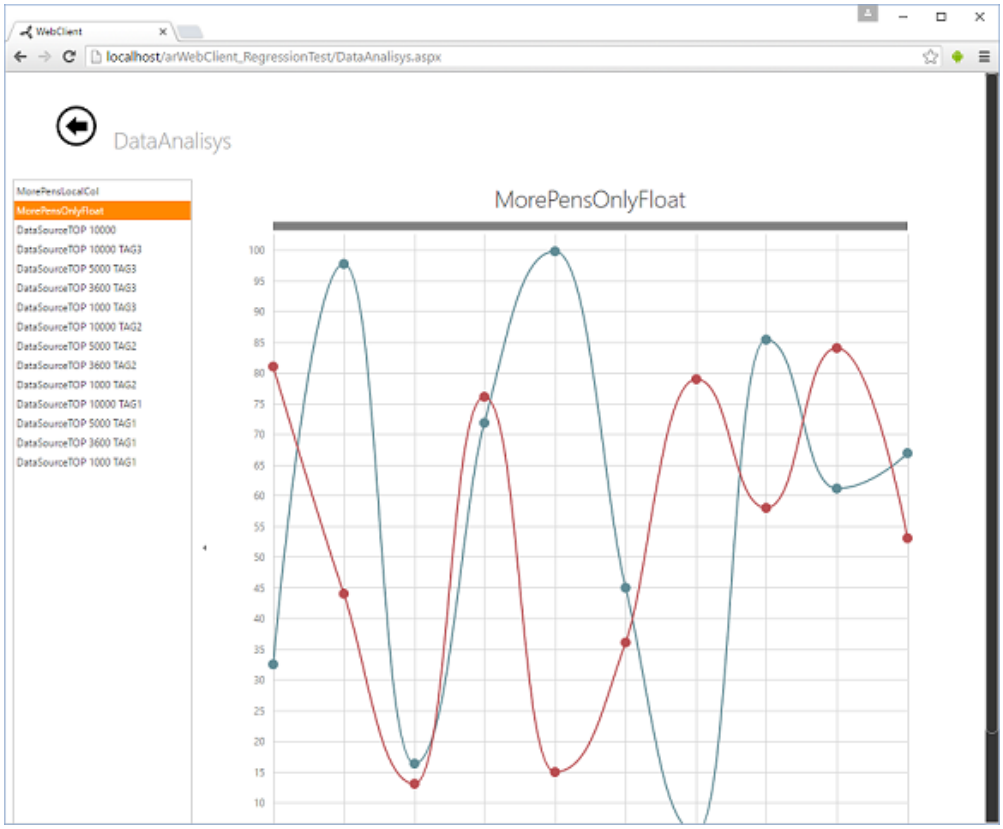


Data Filters can be created within this page in order to export them in various compatible formats.

Data Analysis

The Data Analysis objects are managed in the same way as described above but with 2 exceptions that need to be considered during the deployment procedures:

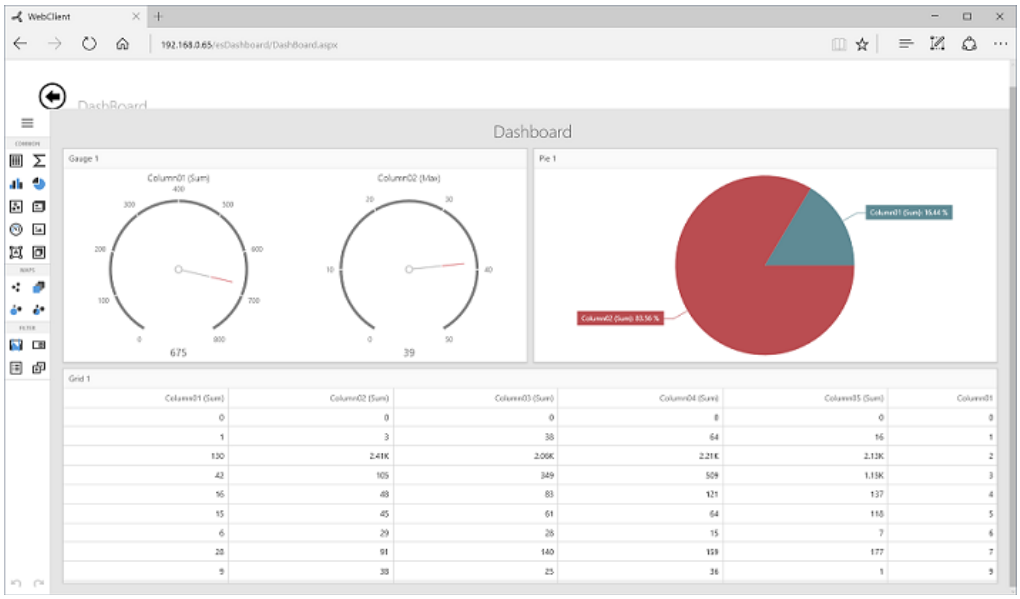
- the first parameter in the select field (where the query is to be inserted) must always be a time column.
- The fields after the query will be used to represent the chart's curves.



There are two calendar controls underneath the 'Date range picker' caption at the top of the page which can be used to apply a filter on data relating to the DataAnalysis within the time range defined in the initial query.
Data is loaded when the fetch button is pressed.

Dashboard

The DashBoard object is used to create a page or pages where different objects can be inserted (eg. Chart, Grid and Gauges) and connected to data sources. These Dashboards can also saved and reused for further use.



The option to create/edit new Dashboards has been included with the program's 3.1 version. By means of making a simple modification to the web.config file, as indicated in the

examples below, new Dashboards can be created or the option allowing this can be disabled.

```
<appSettings file="">
  <clear />
  <add key="Uri" value="C:\inetpub\wwwroot\arTestDashboard\Project\arTestDashboard.UFProject" />
  <add key="ClientSessionName" value="arTestDashboard" />
  <add key="RefreshPollingTime" value="100" />
  <add key="RefreshPollingTimeCount" value="10" />
  <add key="DelayBroadcaster" value="250" />
  <add key="SessionTimeout" value="60" />
  <add key="ConcurrentRenderingPipeline" value="5" />
  <add key="DisableStaticOptimization" value="False" />
  <add key="DisableCreateNewDashboard" value="False" />
  <add key="Theme" value="Blend" />
  <add key="ValidationSettings:UnobtrusiveValidationMode" value="None" />
  <add key="ChartImageHandler" value="storage=file;timeout=20;dir=c:\TempImageFiles\;" />
  <add key="StaticOptimization" value="False" />
</appSettings>
<connectionStrings>
  <remove name="LocalSqlServer" />
  <add name="LocalSqlServer" connectionString="Server=(local);Database=Movicon.Membership;Integrated Security=SSPI" />
</connectionStrings>
```

*<add key="DisableCreateNewDashboard" value="False" /> in this case it will be possible to create new d*ashboards

<add key="DisableCreateNewDashboard" value="True" /> disables the possibility to create new dashboards

1.3.8. Client Users

The Platform.NExT supports a security management for Client users using a user authentication logon procedure. This means that if the Project Server has been set with this security management any users wishing to access the project from the Web Client will have to logon according to User rights and password defined in the Server project. In order to do this the "**Enable User Manager**" property from the project's User Management must be enabled.

In this case the project will be deployed configuring the web.config as indicated:

```
<authorization>
    <allow users="?" />
  </authorization>
```

which denies access to website to all unauthorized users.

While if "False"

```
<authorization>
    <allow users="*" />
</authorization>
```

allows access to all users.

Warning: the user level management currently behaves differently on the desktop client: if a user has a lower level than the one set on the HTLM5 Client, the command objects that require a higher authentication lever will not be made visible.

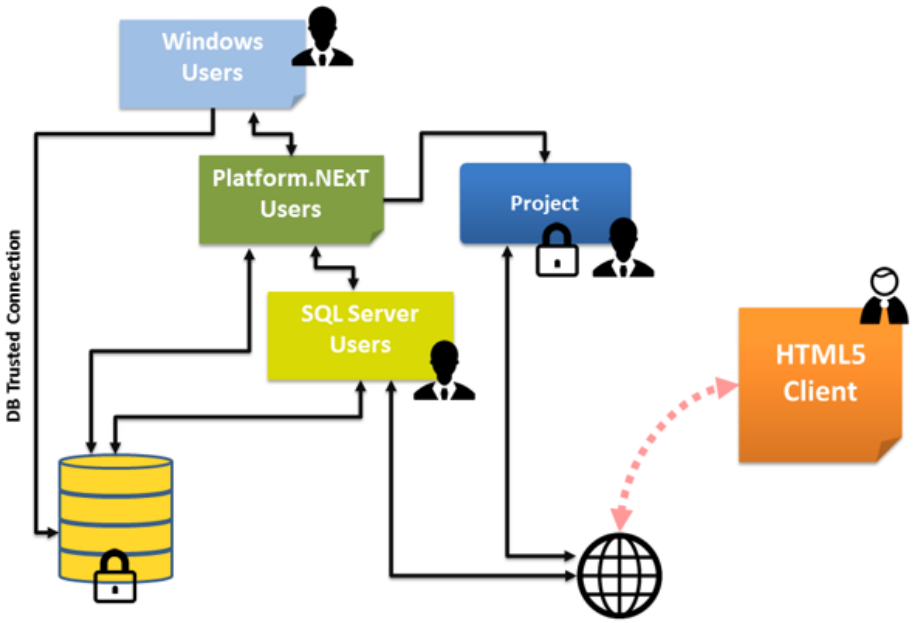
User Management for database access

In addition to the Users management for accessing project command defined on the server, it may also be necessary to configure and define access to databases from the Web Client side. As web application performs direct access to the Database without receiving data from the server, the web application must be "acknowledged" and authorized by the SQL Server database.

It is for this reason that when a web application is deployed, the Pool and Identity are defined to determine the web application's execution and the user identities must be among those which possess user rights and access right to the SQL Server instance.

In order to view the contents of a SQL Server database from the Movicon Server project on the Web Client side the web application will need to be run within a Pool and Identity context recognized and present the SQL Server DB.

See the parameters that can be set when deploying the project from the Server on the web.



This shows the main concepts of the user management in applications.

After having enabled the user management, you can use the web to register as a new user by using the register link and access the project pages to display them.
The new user is created with the predefined settings with user level = '0' and with the bit mask completely deseleted.
If you wish to inhibit the creation of new users by registering them over the web, you can modify the web.config as follows:

<system.web>

<authorization>
<allow users="User name to be authorized"/> //several users can be inserted to allow them access by separating their names with a comma
<deny users="*"/> // this denies access to those users that have not been explicitly indicated on the list beforehand.
</authorization>

Once the web.config file is protected and inaccessible by web, it will not be possible to modify it with the web client.

1.3.9. Client Errors

In event of disconnection towards a "**<ObjectName> - OPC UA Session Keep Alive bad status>**" error message will be reported in the objects displayed on the Web Client.
This error indicates that the Server connection is not available. Therefore you will need to check whether the Server address is reachable and that the project on the Server has been started up in runtime correctly.

INDUSTRIAL AUTOMATION SOFTWARE

progea

Progea Srl
Via D'Annunzio, 295
I-41123 Modena
info@progea.com
Tel +39 059 451060

Progea International SA
via Sottobisio, 28
6828 Balerna (CH)
international@progea.com
Tel +41 91 96 76 610

Progea Deutschland GmbH
Marie-Curie Str., 12
D-78048 VS Villingen
info@progea.de
Tel +49 (0)7721 99838 0

Progea North America Corp.
2380 State Road 44, Suite C
Oshkosh, WI 54904
info@progea.us
Tel. +1 (888) 305-2999