

# Table of Contents

<b>Chapter I</b>	<b>Introduction</b>	<b>1</b>
<b>Chapter II</b>	<b>Project Manager</b>	<b>1</b>
<b>Chapter III</b>	<b>Menus</b>	<b>2</b>
1	Introduction .....	2
2	File .....	3
	Introduction .....	3
3	Edit .....	3
	Introduction .....	3
4	Project .....	4
	Introduction .....	4
	Execute .....	4
	New .....	4
	Duplicate .....	5
	Delete .....	5
	Rename .....	5
	Default .....	5
	Build protected copy .....	5
	Build source copy .....	5
	Export .....	6
	Import .....	7
	Publish Java Web Client and Smart app .....	8
	Test Java Web Client .....	11
	Browse project folder .....	11
5	Device library .....	12
	Introduction .....	12
	Export project as a device .....	12
	Import device as a new project .....	12
6	View .....	12
	Introduction .....	12
7	Help .....	14
	Introduction .....	14
<b>Chapter IV</b>	<b>Configuration</b>	<b>15</b>
1	Project configuration .....	15
2	Options .....	16
3	Channels .....	24
4	Devices .....	25
5	Access groups .....	25
6	Templates .....	26
7	Events/Alarms .....	27
8	Multilanguage .....	27

<b>Chapter V</b>	<b>Gates</b>	<b>28</b>
1	Gates .....	28
<b>Chapter VI</b>	<b>Code</b>	<b>30</b>
1	Code .....	30
<b>Chapter VII</b>	<b>Recipes</b>	<b>32</b>
1	Recipe models .....	32
<b>Chapter VIII</b>	<b>Reports</b>	<b>34</b>
1	Reports .....	34
2	TXT reports .....	37
3	RTF reports .....	39
4	DAT reports .....	43
<b>Chapter IX</b>	<b>Template</b>	<b>45</b>
1	Template .....	45
<b>Chapter X</b>	<b>Images</b>	<b>47</b>
1	Images .....	47
<b>Chapter XI</b>	<b>Keyboards</b>	<b>48</b>
1	Keyboards .....	48
<b>Chapter XII</b>	<b>Java web templates</b>	<b>50</b>
1	Java web templates .....	50
<b>Chapter XIII</b>	<b>Smart apps</b>	<b>51</b>
1	Smart apps .....	51

# 1 Introduction



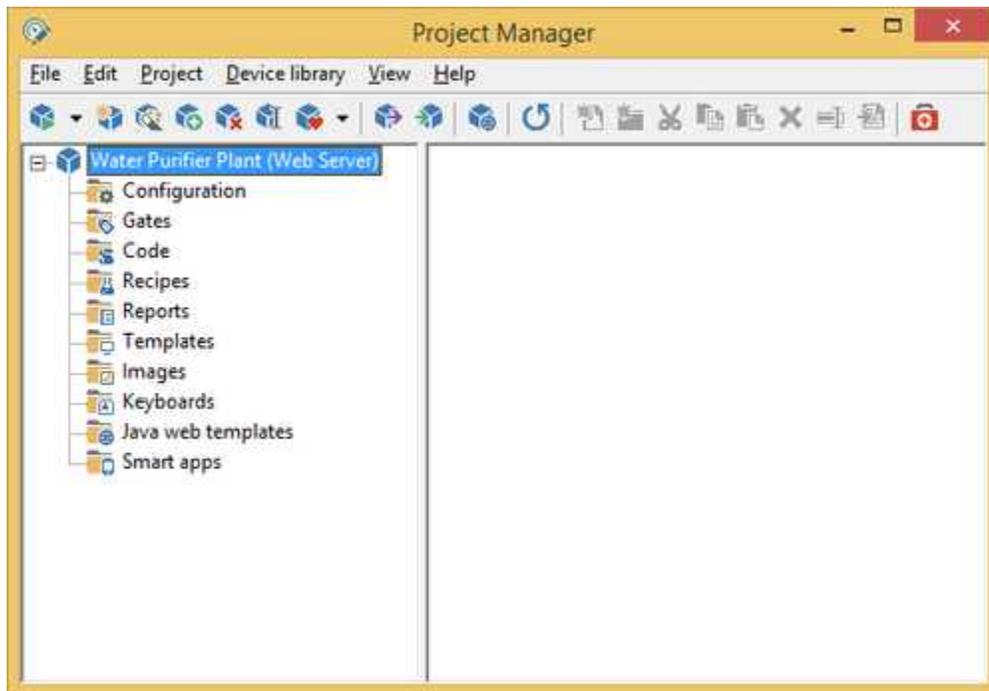
A supervision project is a software product that allows you to monitor and interact with a particular process in industrial and civil environment; eg. control processing steps of an industrial product, management of a home automation system, monitoring of the water levels of a water pipeline, supervision of a photovoltaic system, etc.

*Project Manager* lets you create and manage one or more projects of supervision, by providing the tools to:

- Communicate with the process (through the selection of specific communication protocols).
- Define the list of variables to be exchanged with the process (through the instrument *Gate Builder*)
- Create the Synoptics to view the status of the process and to allow the user to interact with it (via *Template Builder* and *Smart App Builder* tools)
- Create automatic procedures to control the process (using the tool *Code Builder*).
- Configure reports and production recipes

## 2 Project Manager

*Project Manager* main window is the following:



On the left of the screen are listed all the projects present with all their folders, while on the right are displayed the components of the selected folder.

It is possible to: create, delete, copy, configure the projects in a visual manner.

Each project is compound of following folders:

- **Configuration:** project general configuration
- **Gates:** items list (*Tags*)
- **Code:** any user-defined functions using *Code Builder*
- **Recipes:** project recipes model
- **Reports:** project production reports model
- **Templates:** server application synoptic list
- **Images:** list of images used in the project
- **Keyboards:** definition of virtual keyboards, particularly suitable for applications on touch screen
- **Java web templates :** list of synoptic defined for Java Web Client application
- **Smart apps :** definition of the Smart app connected to the project, particularly suitable for Smartphone and Tablet devices.

## 3 Menus

### 3.1 Introduction

Now we are going to see in detail the functions provided by the menus.

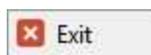


While you are working, there are two ways to access the menus: the first is to use the mouse, the second is to use the *ALT* key in combination with the underlined letter of the menu.

## 3.2 File

### 3.2.1 Introduction

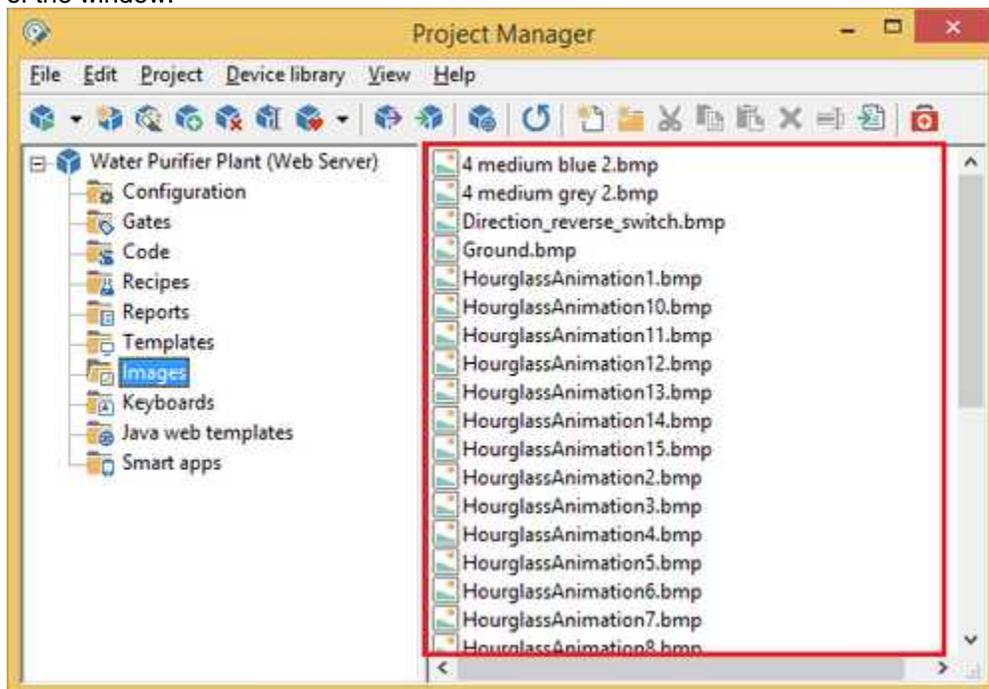
*File* menu contains a single function that allow to exit Project Manager.

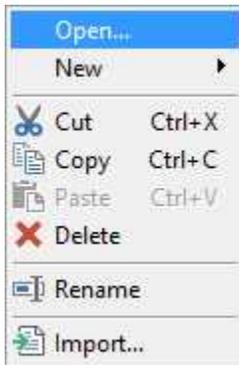


## 3.3 Edit

### 3.3.1 Introduction

The *Edit* menu contains the functions necessary to the manipulation of the selected file in the right part of the window.

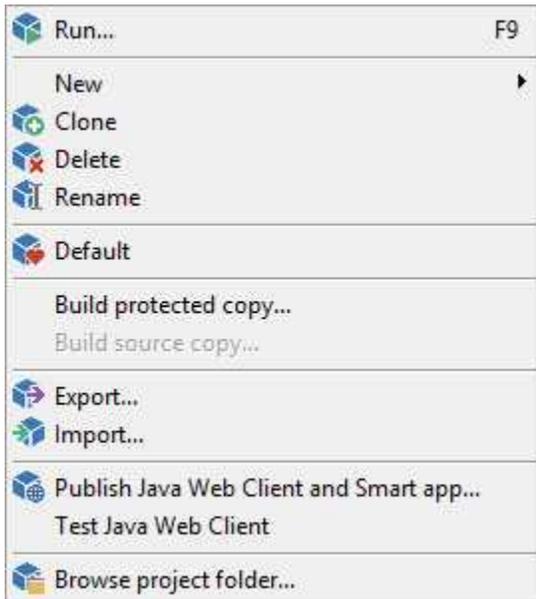




## 3.4 Project

### 3.4.1 Introduction

The *Project* menu contains the functions necessary for handling projects.



### 3.4.2 Execute

Run the selected project.

### 3.4.3 New

You can create new empty projects, that is not configured, selecting the menu item *Project | New*. You will be asked the name of the project that can contain spaces and a maximum length of 255 characters.

You can also create a new project using *Application Builder*: a tool that allows you to quickly and easily create complex applications using devices in the library as basic bricks .

### 3.4.4 Duplicate

To duplicate a project you need to select it and call up the *Duplicate* command from the *Project* menu. It will be necessary to give a name to the duplicated project.

### 3.4.5 Delete

To delete a project select its name and then *Delete* from the *Project* menu. A confirmation of this operation will be required.

### 3.4.6 Rename

To rename a project select its name and then *Rename* from the *Project* menu. Enter the new name of the project.

### 3.4.7 Default

The default project is the one that is executed when you execute the *RunTime* directly, without *Project Manager*. This project is highlighted by a red icon. To make a project the default project, you just need to select its name on the left side of the window, press the mouse right button and select *Default* from the popup menu; you can also select *Default* from the *Project menu*. The opposite procedure is analogous.

### 3.4.8 Build protected copy

With this option it is possible to create a copy of the project protected with a password. I can protect a project to prevent unauthorized altering to templates, code files and gates.

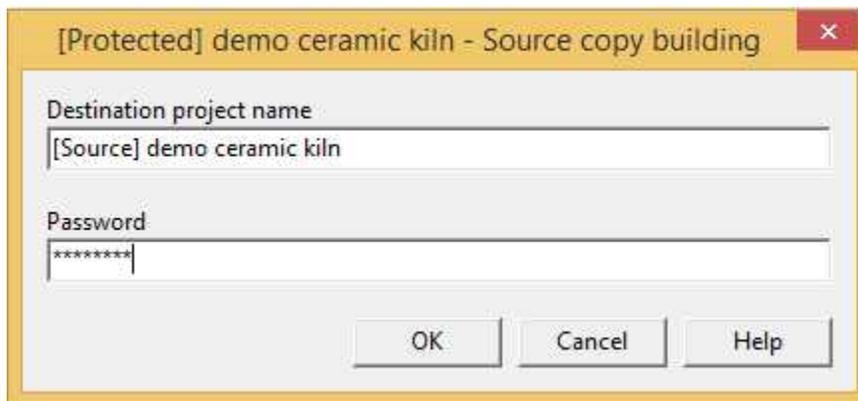


Enter the name of the protected project copy and a password. Press *OK* to start building procedure; if this finishes successfully the window will be closed automatically.

The same password can be used to revert the protection procedure, so to create a source copy from a protected copy.

### 3.4.9 Build source copy

With this option it is possible to create a source copy of the project protected with a password. So encrypted elements (templates, code files and gates) of a protected projects will become editable.



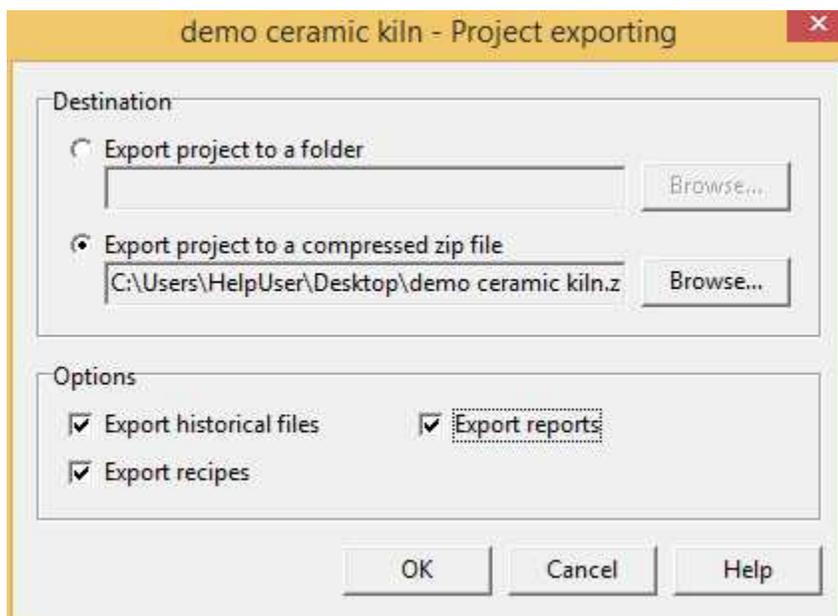
Enter the name of the source project copy and the same password used to protect the project. Press **OK** to start building procedure; if this finishes successfully the window will be closed automatically.

See also: build a protected copy of a project.

### 3.4.10 Export

This dialog allow to export a copy of the project.

Use this feature to make a backup of your projects or to create a copy to deliver to your customers. Exported project can be re-imported using import function.



#### **Destination**

*Export project to a folder:* choose this to export project to a folder; to choose destination folder press *Browse...* button.

*Export project to a compressed zip file:* choose this to export project to a compressed zip file; to choose destination file press *Browse...* button.

#### **Options**

*Export historical files:* flag this option to export historical files.

*Export recipes:* flag this option to export created recipes. Unsetting this option, will be exported only recipe models.

*Export reports:* flag this option to export created reports. Unsetting this option, will be exported only reports configuration.

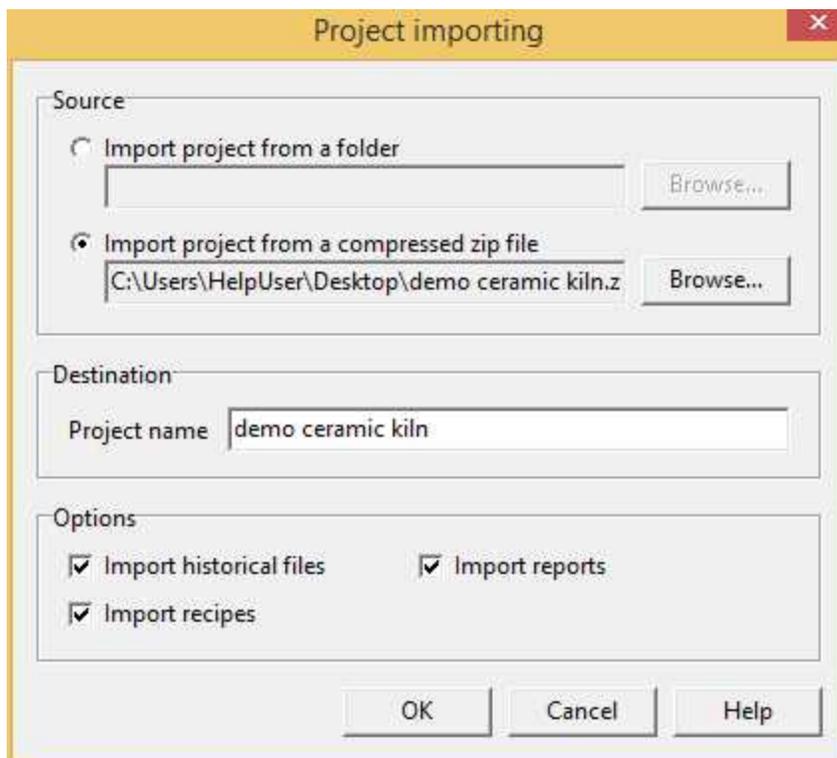
*Compatible with older version:* this flag apply only if you choose to export project to a compressed zip file. Set this flag to create a zip file importable by versions of Project Manager previous to 2.06.64.

If the export procedure finishes successfully the dialog will be closed automatically.

See also: import a project.

### 3.4.11 Import

This dialog allow to import a project previously exported.



#### Source

*Import project from a folder:* choose this to import a project from a folder; to choose source folder press *Browse...* button; folder selecting dialog allow to choose only folders that appear to contain a valid project.

*Import project from a compressed zip file:* choose this to import a project from a compressed zip file; to choose destination file press *Browse...* button. An error message box will be displayed you selected a zip file that appear to not contain a valid project.

**Destination**

*Project name:* system will suggest to you a destination project name based on source folder (or compressed zip file), but you can edit it.

**Options**

*Import historical files:* flag this option to import historical files.

*Import recipes:* flag this option to export created recipes. Unsetting this option, will be imported only recipe models.

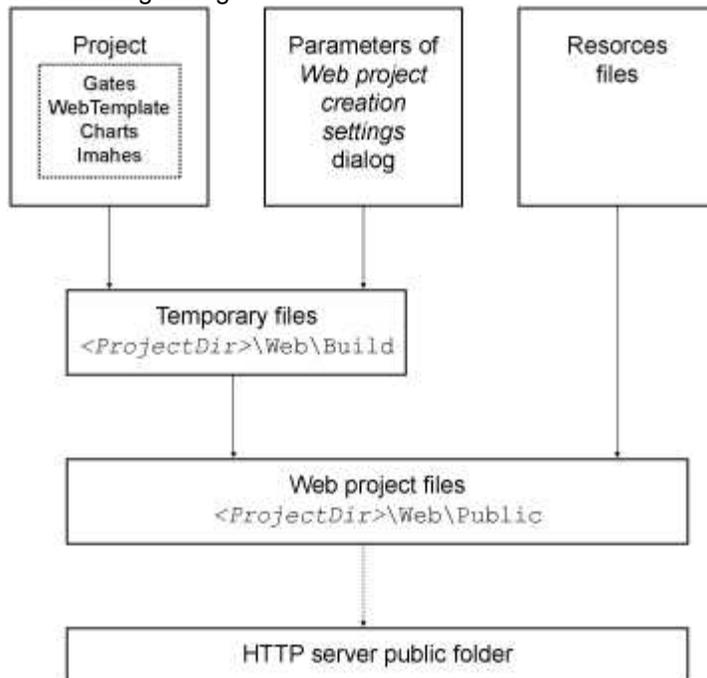
*Import reports:* flag this option to export created reports. Unsetting this option, will be imported only reports configuration and report master files.

If the import procedure finishes successfully the dialog will be closed automatically.

See also: export a project.

**3.4.12 Publish Java Web Client and Smart app**

Web project publication procedure generates some files required to run a web project. These files are created from original project elements (gates, templates, images, ...) and using settings specified in the following dialog.

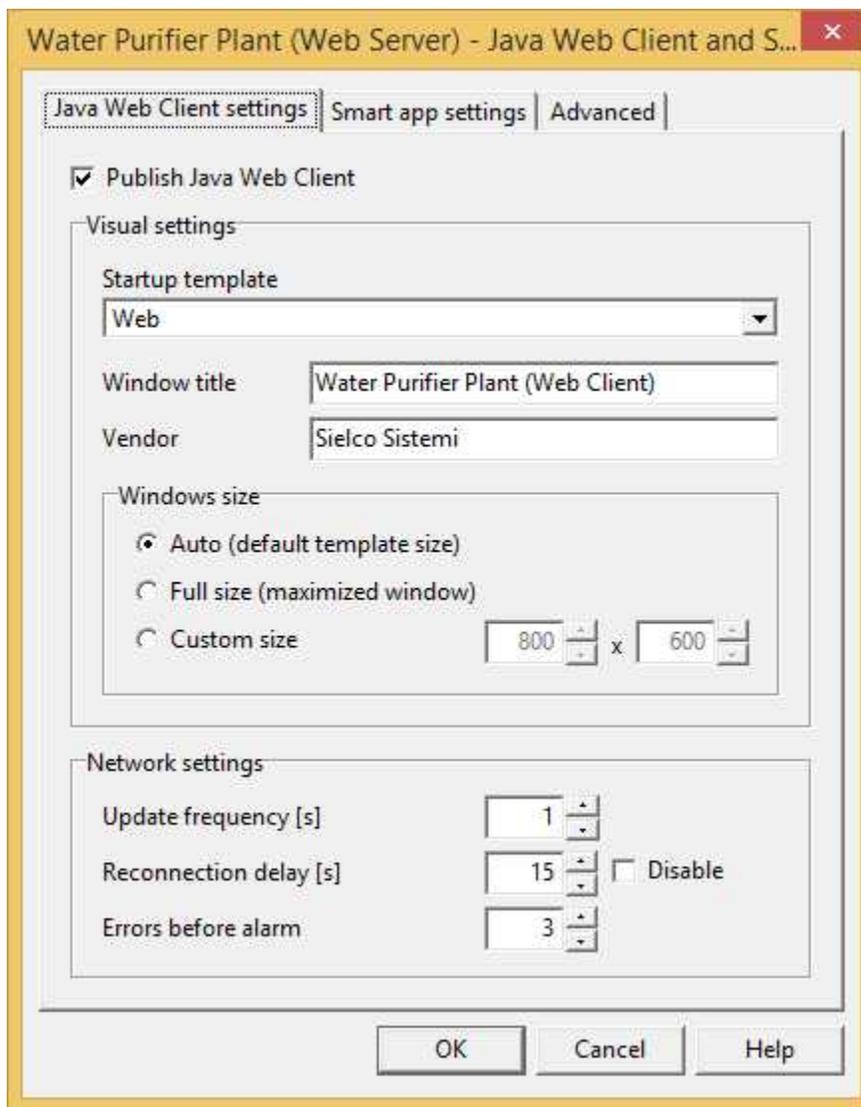


Web project refers to *Java Web Client* application and *Smart app*.

The publication procedure, after verification of the data entered, is launched when you press the OK button of the following dialog.

***Java Web Client settings***

In the next tab you can configure the publication of the Java Web Client.



*Publish Java Web Client:* enabling this option allows the publication of the application *Java Web Client*

*Startup template:* is the template opened by *Java Web Client* at its start. Choose one template using combo box.

*Window title:* is the caption of main window of *Java Web Client*.

*Vendor:* insert your name or the name of the your company. Inserted text will be showed at start of *Java Web Client*.

*Window size:* choose the size of the *Java Web Client* window:

- *Auto (default template size):* window size is automatically set to the size of the template
- *Full size (maximized window):* window is maximized
- *Custom size:* specify size of window (in pixels).

*Update frequency [s]:* *Java Web Client* refreshes data using this frequency (in seconds).

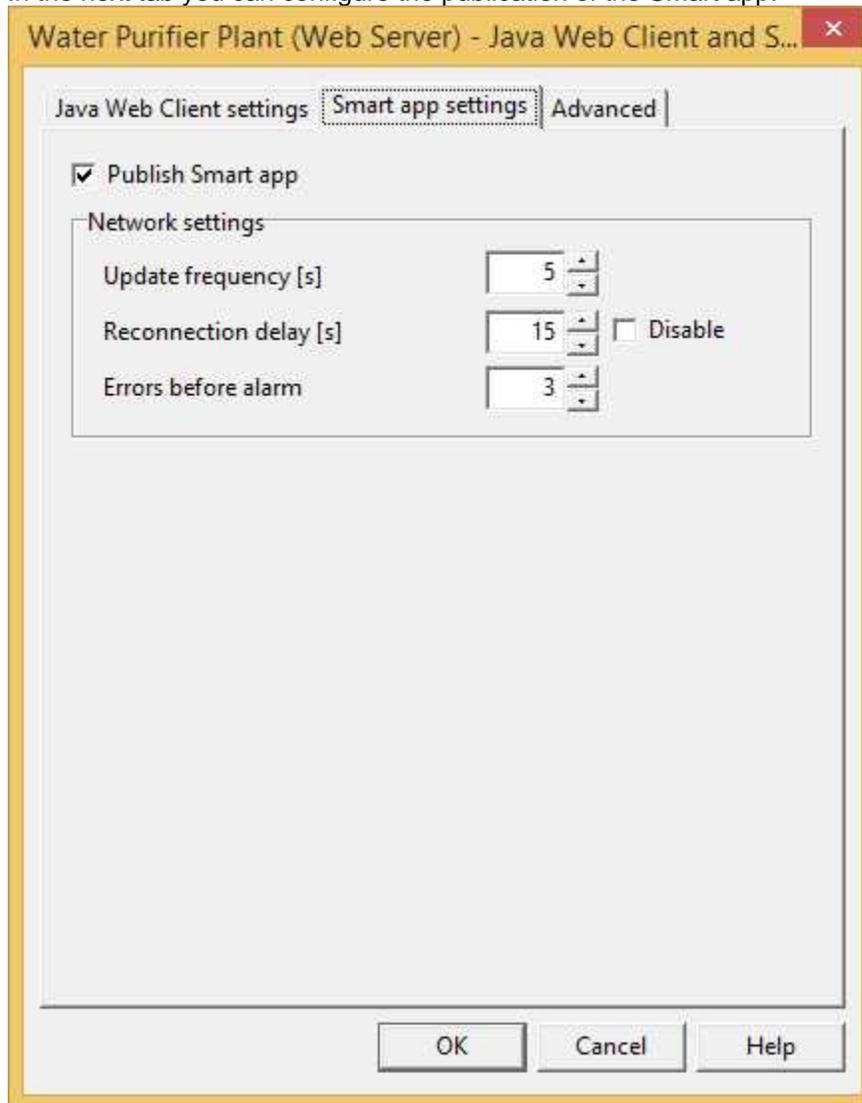
*Reconnection delay [s]:* if reconnection is enabled, use this to specify delay (in seconds) from a connection request and the following.

*Errors before alarm:* specify number of connection errors or read errors allowed. After them an error

message will be displayed.

### Smart app settings

In the next tab you can configure the publication of the Smart app.



*Publish Smart app*: enabling this option allows the publication of the Smart app.

*Update frequency [s]*: Smart app refreshes data using this frequency (in seconds).

*Reconnection delay [s]*: if reconnection is enabled, use this to specify delay (in seconds) from a connection request and the following.

*Errors before alarm*: specify number of connection errors or read errors allowed. After them an error message will be displayed.

### Advanced

In the next tab you can configure some publication parameters that apply to both the Java Web Client and Smart app.



*Don't overwrite web static files:* setting this flag, creation procedure will avoid to overwrite `index.html` and `splash.png` files. Set this flag if you personalized these files, otherwise procedure will overwrite them with the standard ones.

*Clean temporary build files after publishing:* usually this flag is set. Remove flag only for debug purpose: temporary files may waste your hard disk space.

### 3.4.13 Test Java Web Client

Run on the local machine the Java Web Client .  
It is necessary that the server application is active.

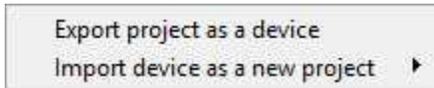
### 3.4.14 Browse project folder

Use Windows Explorer to show the selected project folder.

## 3.5 Device library

### 3.5.1 Introduction

The Device Library menu contains the functions necessary for the manipulation of device library.



### 3.5.2 Export project as a device

With this function you can copy the project in the devices library and so it can be used as the basis brick in the construction of more complex applications using *Application Builder*. (To recall *Application Builder* select the menu item "*Project | New | Project using ApplicationBuilder ...*")

The expansion and customization of the library can be achieved by creating simple applications, each of which refers to a specific model of device and, after having tested the correct functioning, copying it to the library.

To create a new device you have to follow the following steps:

- By *Project Manager*, create a new project by assigning the name of the device.
- Specify on channel 1 the communication protocol and its configuration.
- Through *Gate Builder* define all gates of the device, taking care not to insert in the "Gate ID" a name longer than 17 characters (the maximum length for this field is 20 characters, but *Application Builder* during the construction of the application will fit into this field a prefix of three characters consisting of channel number \* 100 + device number, so you need to reserve the space for these three characters overcoming precisely the length of 17).
- By *Project Manager* to define all the templates of the device.
- Test the project making it communicate with a real device.
- Copy the project to the library using this menu item.

### 3.5.3 Import device as a new project

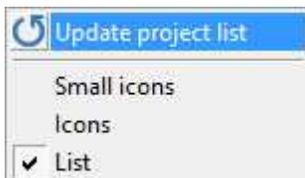
Import in the list of projects a device from library.

After the device has been imported, you will see it in the list of projects shown on the left window of the *Project Manager*: at this point it can be edited and saved back into the library.

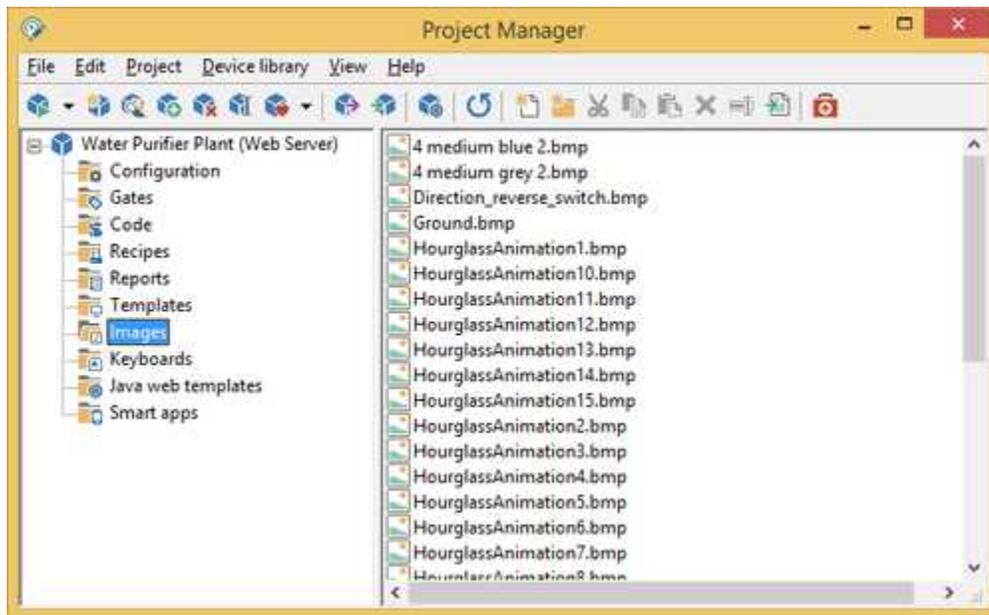
## 3.6 View

### 3.6.1 Introduction

The *View* menu allows you to force a re-reading of the list of projects, and to set preferences on the file display mode.







## 3.7 Help

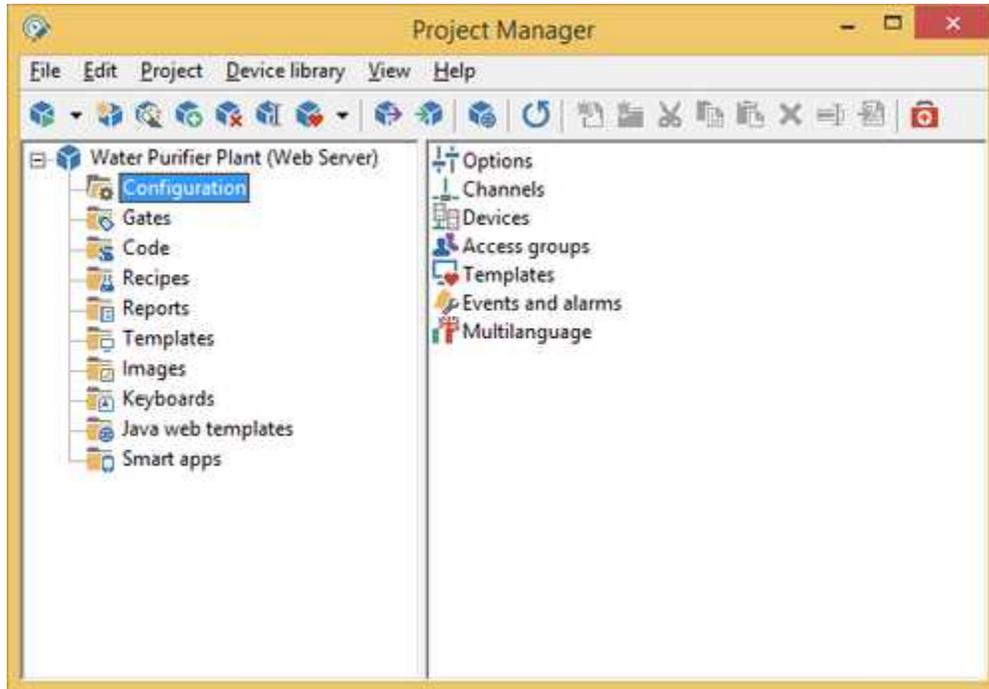
### 3.7.1 Introduction

If you need help it can be useful to read the guide or the contextual help, both accessible from the Help menu.



## 4 Configuration

### 4.1 Project configuration



This project folder has the following configuration elements:

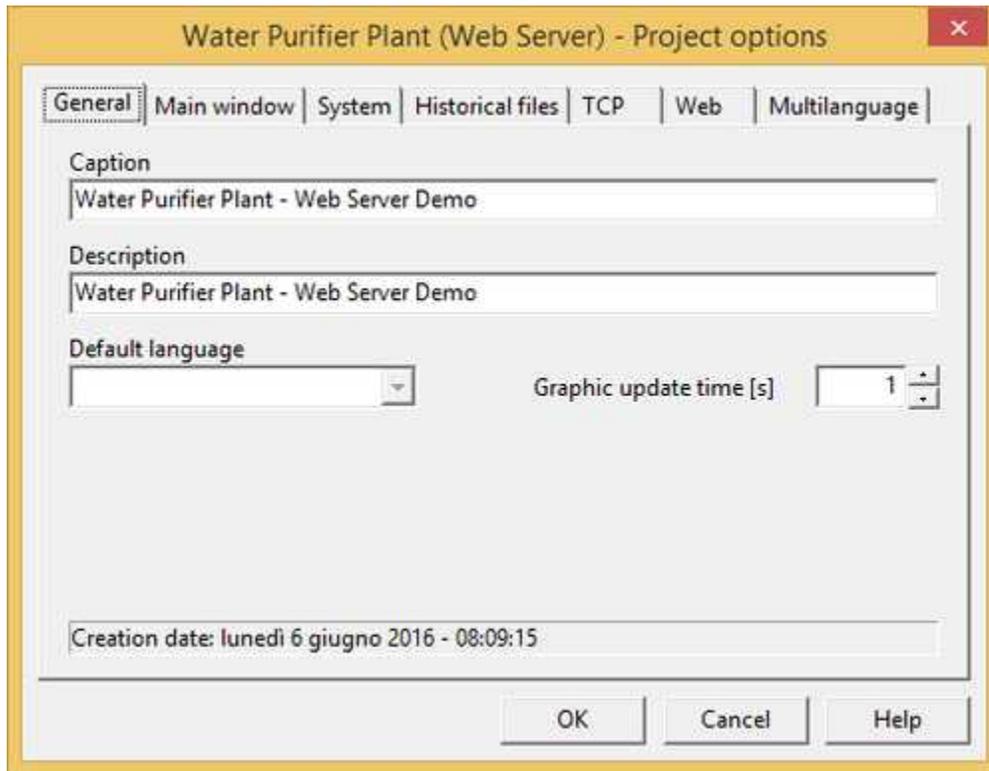
-  Options
-  Channels
-  Devices
-  Access groups
-  Template
-  Events/Alarms.
-  Multilanguage

To configure or modify these elements you just need to double click on the icons to access the corresponding dialog boxes.

## 4.2 Options

The options dialog box allows to configure some general features of the project.

### General page options



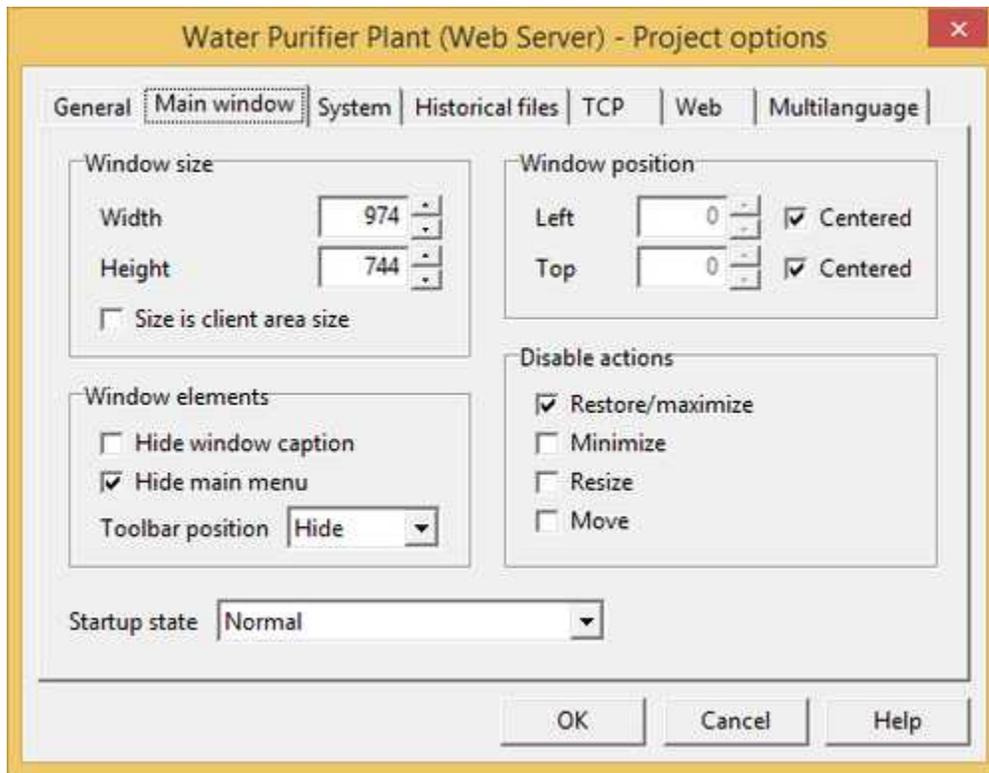
*Caption*: the caption that will be shown during run-time phase.

*Description*: simple description of the project.

*Default language*: the default language that will be used during run-time phase.

*Alarm bar text align*: alignment of the red alarm bar at the bottom of the window.

### Main window page options



This page allows to set appearance of the main window of *Runtime*.

### **Window size**

This is the default size of the main window in normal state (thus when it is not maximized or minimized). These controls may be disabled if there no way to display main window in normal state (for example, using other settings of this page, if you choose to start *Runtime* window in maximized mode and to disable window restore). Note that window contents (templates and other child windows) are not scaled according specified size; if templates are bigger than window size, scroll bars will appear.

*Width*: width (in pixels) of the main window when it is in normal state.

*Height*: height (in pixels) of the main window when it is in normal state.

### **Window position**

This is the default position of the main window in normal state (thus when it is not maximized or minimized). These controls may be disabled if there no way to display main window in normal state (for example, using other settings of this page, if you choose to start *Runtime* window in maximized mode and to disable window restore).

*Left*: x-coordinate of the screen where will be placed the main window (related to its top-left corner).

*Top*: y-coordinate of the screen where will be placed the main window (related to its top-left corner).

*Centered*: flag this options to center the window on the screen horizontally, vertically or both.

### **Disable actions**

These check boxes allow to disable user actions. Some options can be disabled according to other

settings of the page.

*Restore/maximize*: deny users to restore main window when it is maximized and to maximize main window when it is in normal state.

*Minimize*: deny users to minimize main window.

*Resize*: setting this flag users can not resize normal main window when it is in normal state.

*Move*: setting this flag users can not move normal main window when it is in normal state.

### **Hide elements**

Use these options to hide some elements of the main window. Hiding default elements to gain space for templates or to prevent users to recall unwanted commands.

*Main menu*: hide menu bar of *Runtime* main window.

*Lower buttons bar*: hide lower buttons bar of *Runtime* main window.

*Caption bar*: hide caption bar of the main window (caption bar is the upper bar of the window, that contains icon, caption and minimize, maximize and close buttons). Hiding caption some actions will be disabled automatically (users can not restore/maximize, minimize, resize or move main window).

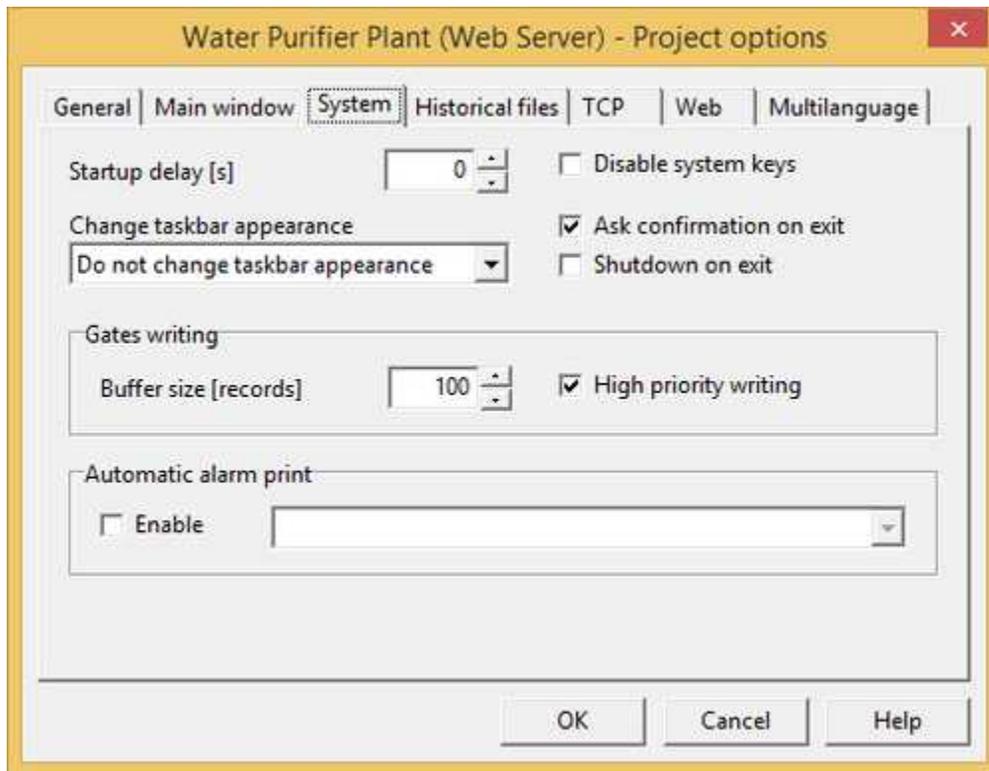
*Templates objects popup menu*: context popup menu of templates objects is disabled.

### **Startup state**

Define startup state of the *Runtime* main window.

- *Maximized*: *Runtime* main window will be opened automatically maximized.
- *Normal*: *Runtime* main window will be opened in normal state. Its dimensions are that specified by *Window size* group controls and it will be placed at coordinates specified by *Window position*.
- *Depending on screen resolution*: startup state will be determined runtime: if screen is bigger than *Window size* main window will be opened in normal state, otherwise it will be opened maximized.

### **System page options**



*Startup delay*: specify the number of seconds to wait before to start communication after *Runtime* startup. When *Runtime* is defined to automatically run on Windows startup, it is better to set this parameter to 10 seconds to allow Windows to initialize all drivers.

*Change taskbar appearance*: this allows to automatically change the taskbar appearance during run-time phase, thus to prevent taskbar to occupy screen space or overlap *Runtime* windows. You can choose to maintain your current system settings (*Do not change taskbar appearance*), or to set task bar to auto-hide itself when you click on other windows (*Set task to auto-hide*) or to not stay on top and to prevent taskbar to overlap other windows (*Set taskbar to not hide other windows*). Choose the latter if you want to prevent users to switch to different programs when *Runtime* is running, because *Runtime* window will cover completely taskbar. Choose auto-hide to gain space on screen without denying the ability to recall taskbar.

Please attention that *Runtime* changes are system-wide and it reset taskbar appearance to its original state at program exit. If *Runtime* can not exit gracefully (e.g. due to power fail), original taskbar settings will not be restored. The easiest method to recall a "lost" or hidden taskbar is to press a Windows key or CTRL+ESC. To manually restore appearance right click on a clear area of taskbar and select *Properties* from the context menu.

*Change taskbar appearance* option is not supported in Windows 98 and Windows 2000.

*Disable system keys*: if you check the checkbox, system keys will be disabled, so users cannot switch to a different program when *Runtime* is running. Disabled system keys are: ALT+TAB, ALT+ESC, Windows keys and CTRL+ALT+DEL (the latter only in Windows 98).

*Ask confirmation on exit*: enable/disable ask of a confirmation before closing *Runtime*.

*Shutdown on exit*: enable/disable automatic PC shutdown after closing *Runtime*.

### **BSL writing**

*Writing buffer size*: it is the size of the BSL writing queue. All writing requests to device will be put in this queue. A bigger value may increase performances if your project requires several

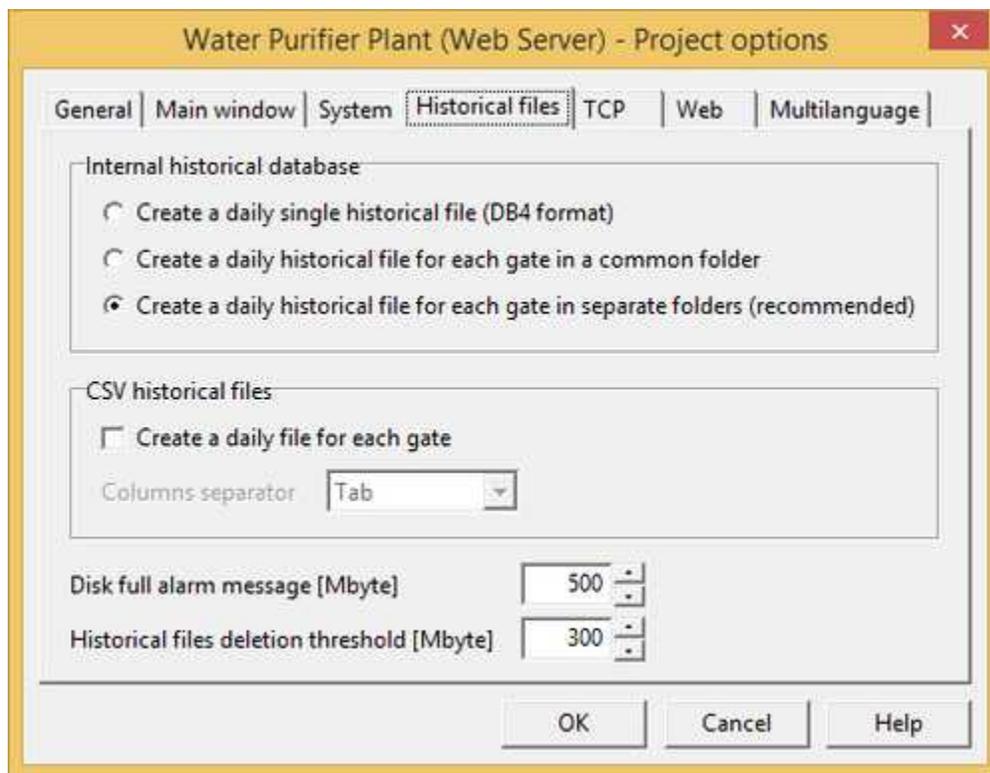
concurrent writing operations, for example if you complex recipes. Each channel has its own queue.

*High priority writing:* if it is enabled, every data that must be send to a device, will be sent immediately, otherwise it will be sent after a complete read sample cycle.

### **Automatic alarm print**

*Enable:* if it is checked, every time that an alarm became true, it will be sent to the selected printer device.

### **Historical files page options**



Historical files contains records of gate value variation of the gates defined as "record on historical file" in Gate Builder. They are saved in DBTABLES subdirectory of the project. More in details, numeric and compound gates are saved in DBTABLES\NUMERIC folder, digital gates are saved in DBTABLES\DIGITAL folder and string gates are saved in DBTABLES\STRING folder.

The content of these folders is organized according with the saving mode selected, as follow.

### **Internal historical database**

*Create a daily single historical file (DB4 format):* if this option is checked, the software will create a single daily historical file for each kind of gate in DB4 format. This mode is supported to grant compatibility with previous application or if you want to elaborate a DB4 historical file with an external software that is able to read this kind of file.

*Create a daily historical file for each gate in a common folder:* if this option is checked, the software will

create a daily historical binary file for each gate defined as to be "recorded on file". This option will increase the historical charts view speed performance. All files are saved in the same folder. If you need to save a large number of gates it is better to use the next option that separate historical files in several subfolders, one for each gate.

*Create a daily historical file for each gate in separate folders (recommended):* if this option is checked, the software will create a daily historical binary file for each gate defined as to be "recorded on file". This option will increase the historical charts view speed performance. Historical files will be divided in subfolders, one for each gate.

### **CSV historical files**

*Create a daily file for each gate:* if this option is checked, the software will create a daily historical CSV file for each gate defined as to be "recorded on file". This kind of file also known as "Formatted text" can be easily imported and manipulated from other software like spreadsheet. Each record of this file is compound of two fields: date and time field and value field. Each record is saved into the file when the gate value changes. If there is a communication error on that gate, an empty record will be added.

CVS files are saved into the subfolder `DBTABLES` of the project. More in details, numeric and compound gates are saved in `DBTABLES\NUMERIC` folder, digital gates are saved in `DBTABLES\DIGITAL` folder and string gates are saved in `DBTABLES\STRING` folder. The filename has the format

`dd_mm_yy_GateId,NId.dat`

where:

`dd`: day of the month

`mm`: month

`yy`: year

`GateId`: Gate Id of the recorded gate

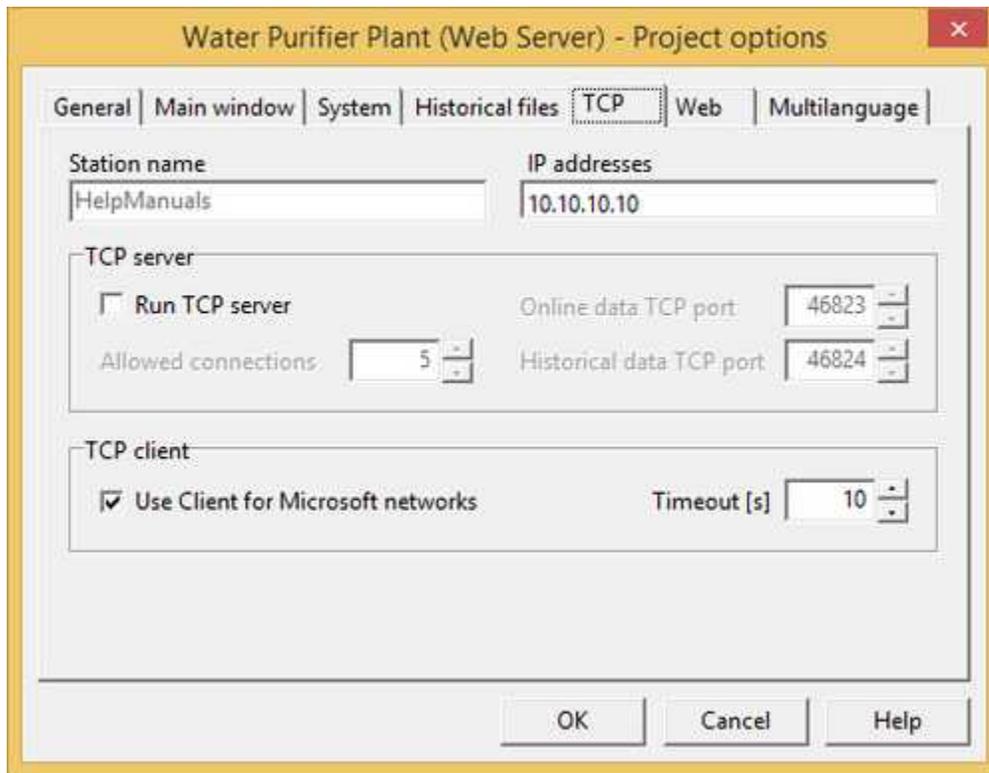
`NId`: N Id of the recorded gate

*Columns separator:* indicate the separator character between two fields in the CSV file.

*Disk full alarm message:* it is the threshold beyond which the program informs the operator that the disk is beginning to be full.

*Historical files deletion threshold:* it is the threshold beyond which the program starts to delete the files of the historical database, starting from the older ones, until it reaches the *Full disk alarm threshold*.

### **TCP page options**



*Station name:* it shows the network ID of your PC

*IP Addresses:* it shows a list of IP addresses assigned to your PC

### **TCP server**

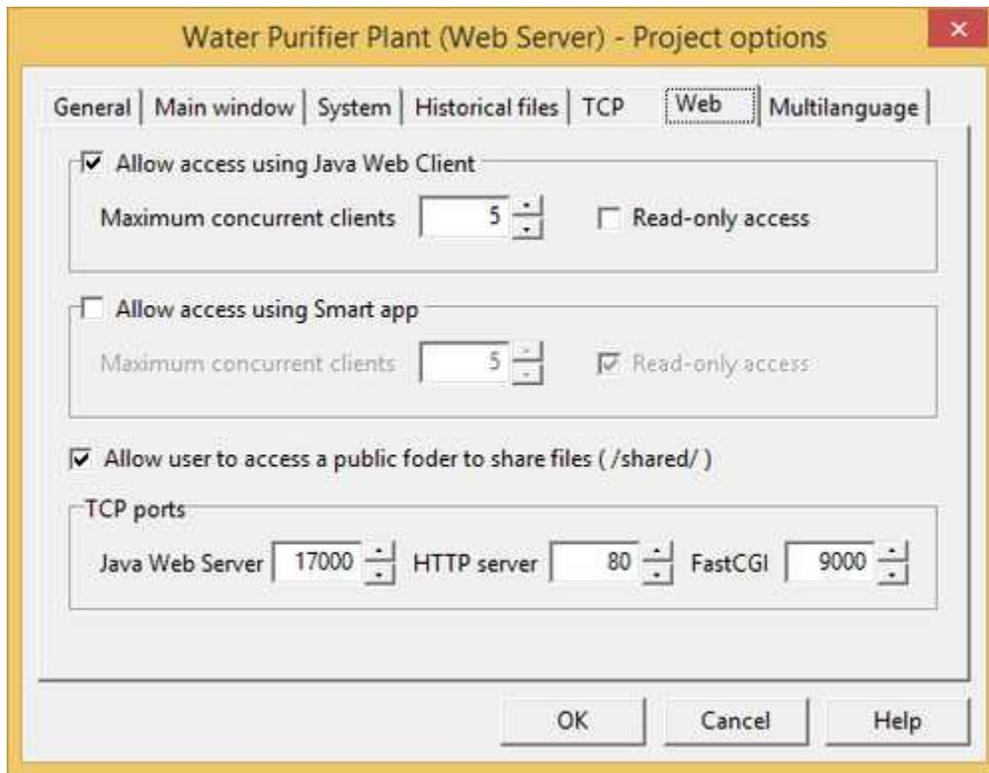
*Run TCP server:* enable *Runtime* to work as a TCP server for other station that uses the same supervision software. Remote stations have to use TCP client protocol. TCP server listen on port 46823 for online data and on port 46824 for historical data.

### **TCP client**

*Use Client for Microsoft network:* with Chart, HistView and OperatorView template objects, it is possible to access historical files on remote server stations. This is possible using TCP/IP protocol. If on the client and server computers is available Client for Microsoft Network protocol, remote historical file access can be faster and efficient. To work properly, on each remote server station, you need to share (with read and write rights) the folder `DBTABLES` of the project. Share name must be `DBTABLES`.

*Timeout:* timeout used by Client for Microsoft networks during access to remote historical files.

### **Web page options**



**Allow access using Java Web Client** : allows the Java Web Client to access to server application.  
*Maximum concurrent clients*: maximum number of concurrent Java Web Client connections permitted.  
*Read-only access*: if this option is enabled, users who connect to the server through the Java Web Client will not be enabled for data modification.

**Allow access using Smart app** : allows the Smart app to access to server application.  
*Maximum concurrent clients*: maximum number of concurrent Smart app connections permitted.  
*Read-only access*: if this option is enabled, users who connect to the server through the Smart app will not be enabled for data modification.

**Allow user to access a public folder to share files (/shared/)** : if this option is enabled then will be enabled the web-based access to the subfolder "Web/Public/shared" where will be the files to be shared with users, such as production reports or historical CSV files.

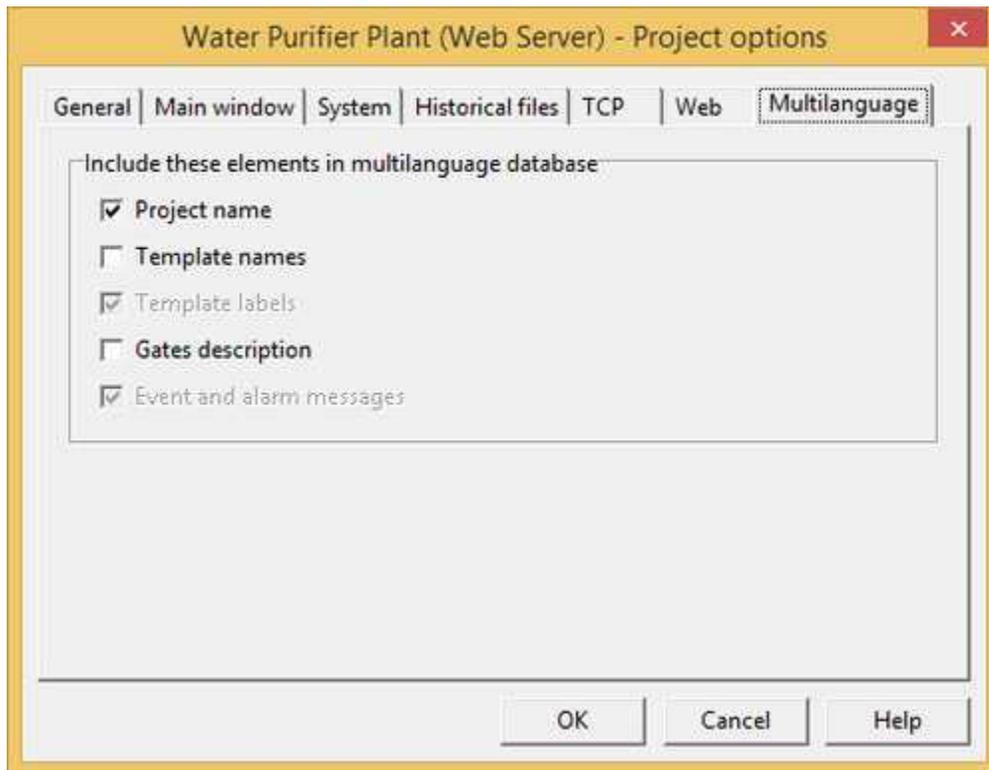
**TCP ports**

*Java Web Server* : specify the listening port of internal Java Web server.

*Server HTTP* : specify the listening port of internal HTTP server.

*FastCGI* : specify the listening port of internal FastCGI server.

### Multilanguage page options



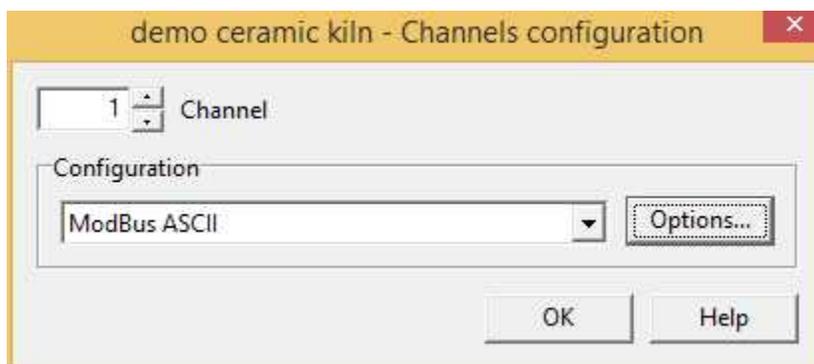
Parameters in this page define which labels must be translated. Translation can be performed running *Multilanguage Editor* from *Project Manager* main window.

Translatable groups of labels are: *Project name*, *Templates name*, *Templates labels*, *Gates description* and *Events/alarms messages*.

## 4.3 Channels

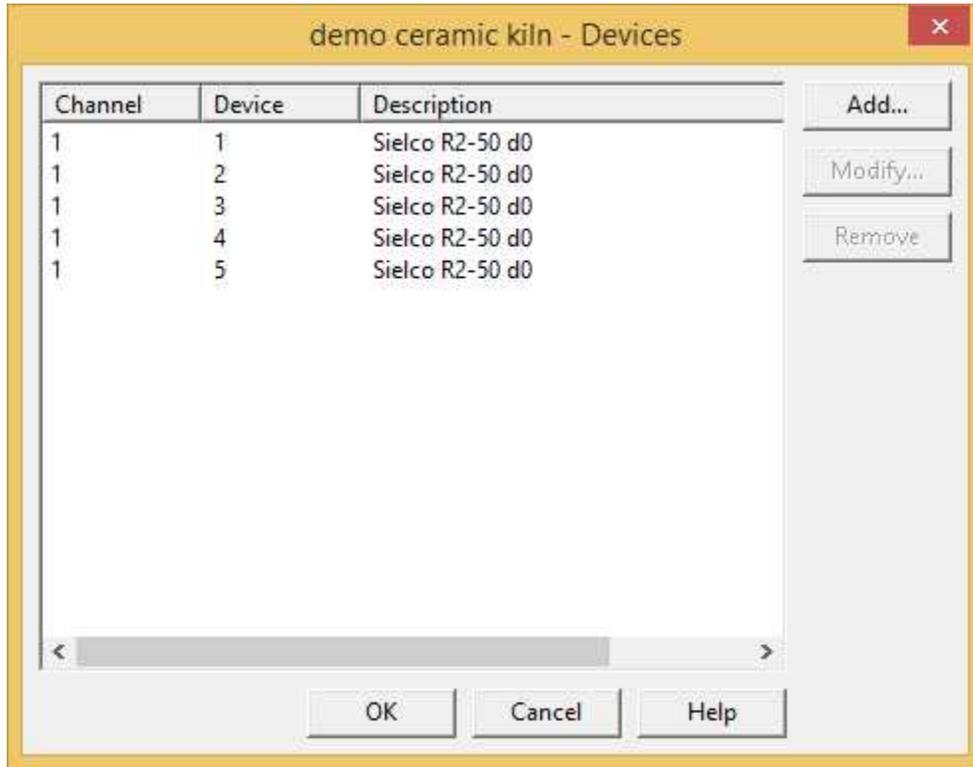
A *Channel* is a communication port used to communicate with the external devices.

*Channel configuration* dialog allows to modify settings of the communication channels. Choose the channel number and then select the communication protocol that you want to use for that channel. Press *Options...* button to configure the selected protocol.



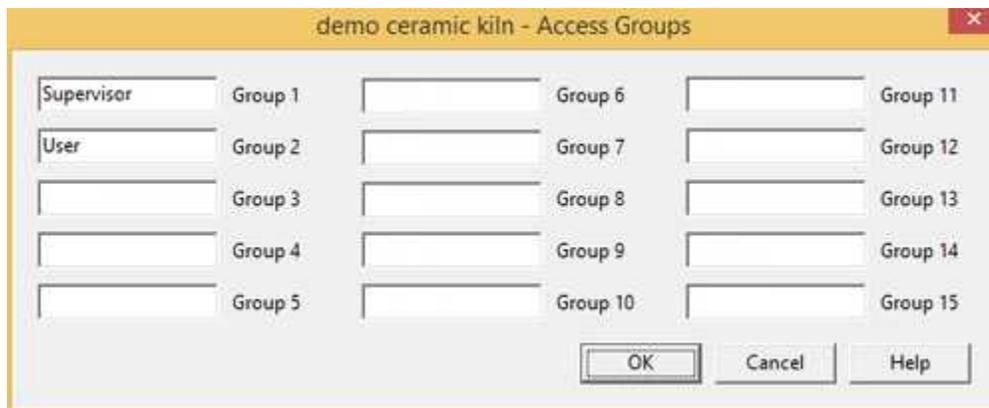
## 4.4 Devices

The software allows to monitor the communication of all the connected devices. These are identified by the communication channel number and by an address. It is possible to describe each device in order to identify it more clearly using the box in figure. This dialog box allows to insert the descriptions of all the used devices, and to modify or delete those already inserted.



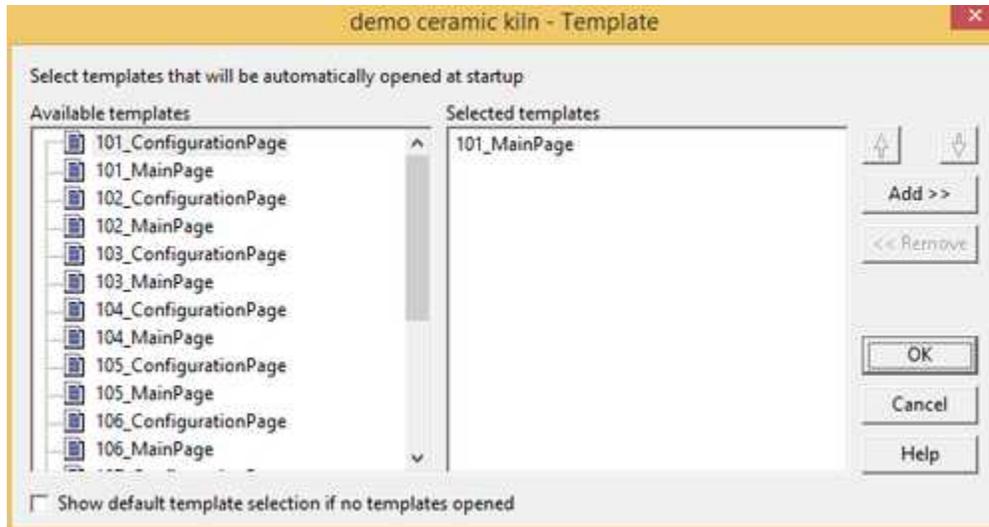
## 4.5 Access groups

Using the box in figure, it is possible to give a name (maximum 12 characters) to each of the 15 access groups, so as to make their management easier and more intuitive.



## 4.6 Templates

### Startup templates page options



*Startup Templates* allows to select templates that must be automatically loaded and opened at *Runtime* startup.

To automatically load and open a template at startup, select it from the left list and press *Add* button; its name will be added to right list. You can use double click on its name in the left list.

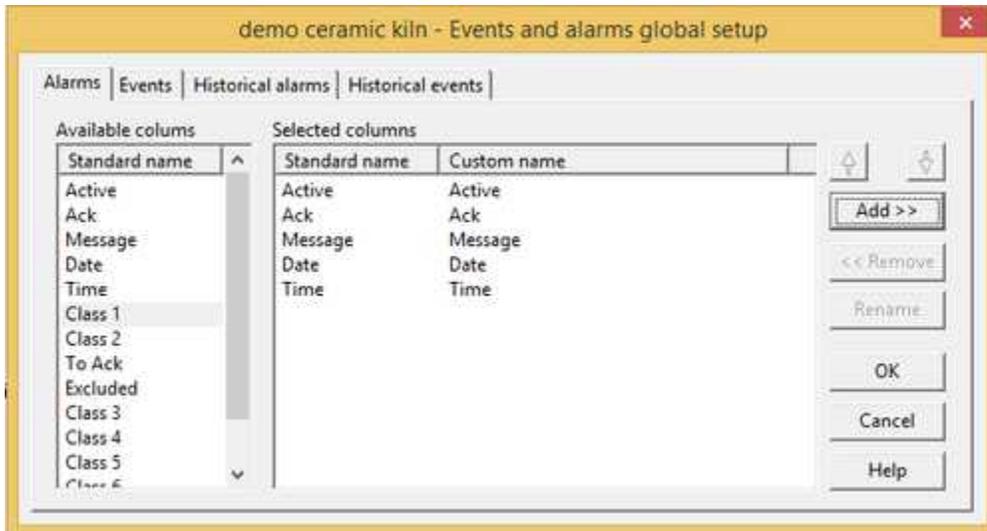
To remove a template from the *Selected templates* list, select its name and press *Remove* button. You can also double click on its name in the right list.

You can select a maximum of 8 templates.

*Show default template selection, if no templates opened*: if no templates are defined to be automatically opened, if you check this box, at *Runtime* startup, a window with a list of all the available templates will be showed.

## 4.7 Events/Alarms

### Alarms, Events, Historical alarms and Historical events page options



These tabs allow to select columns to display into alarms, events, historical alarms and historical events views of *Runtime*.

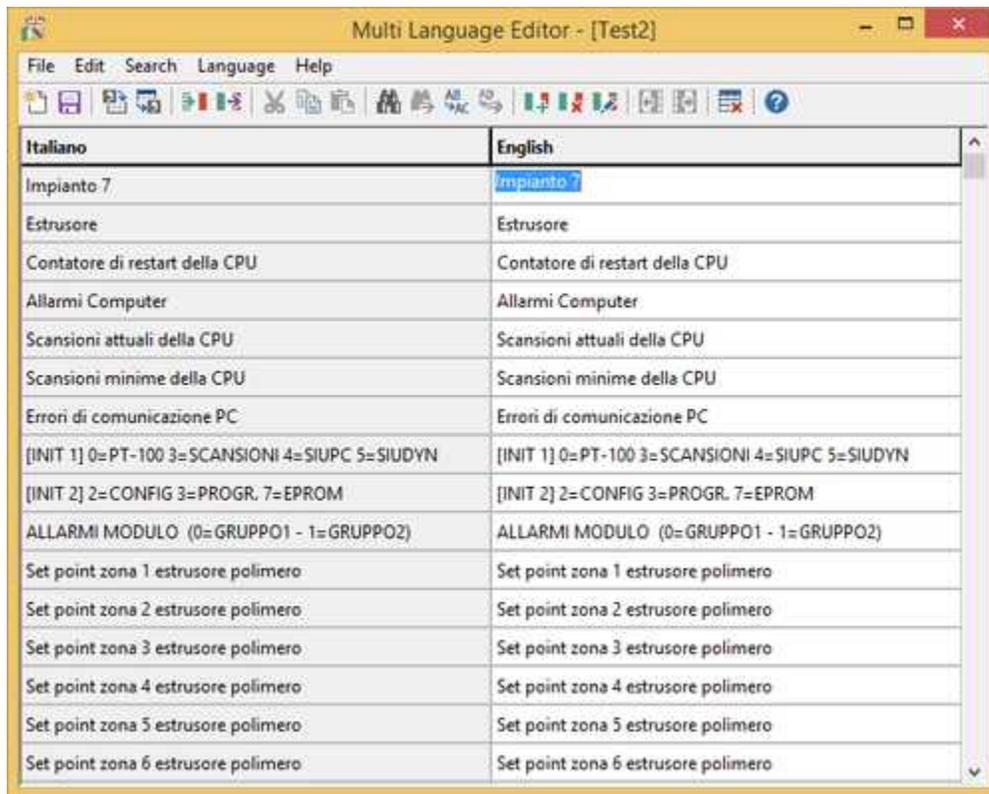
To make a column visible, select it from the left list and press *Add* button; its name will be added to right list. You can use double click on its name in the left list.

To remove a column from the *Selected columns* list, select its name and press *Remove* button. You can also double click on its name in the right list.

Columns that have *Standard name* beginning with "Class" can have generic content, so you can rename them using the *Rename* button.

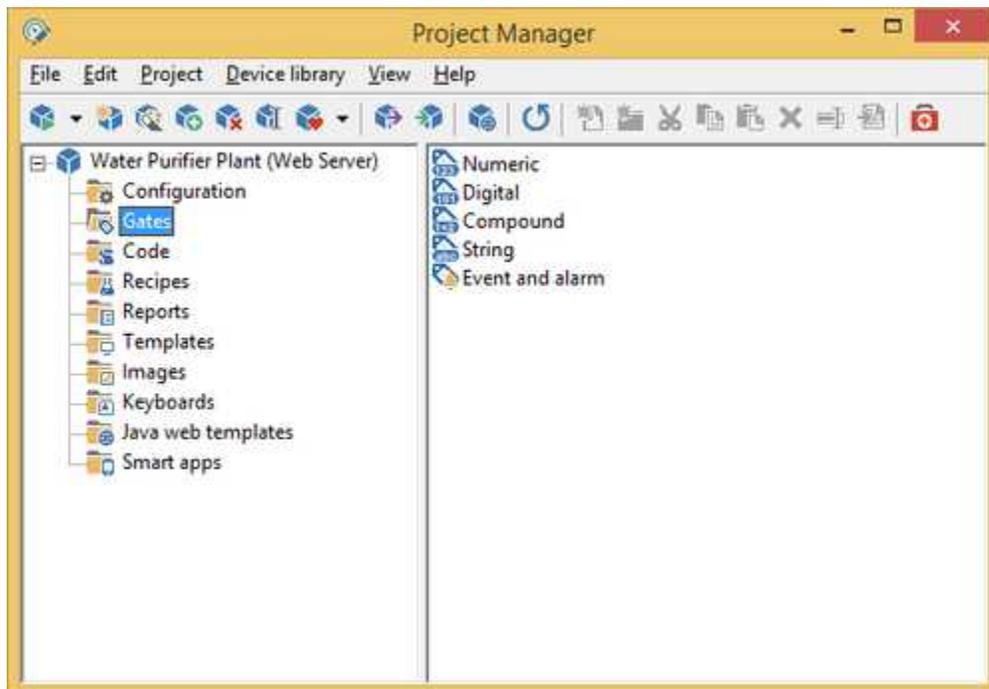
## 4.8 Multilanguage

Often presents the need of having to develop the same application in different languages, for ease of use by the end user. However it is not necessary to generate multiple copies of the same application and then proceed with the translation for each of them; through the instrument *Multi Language Editor* is possible to equip a single application of the multilanguage support and then associate it up to a maximum of 10 languages.



## 5 Gates

### 5.1 Gates



With "Gate" is intended to represent a process variable.

The software manages several types of gates:



**Numerical:** signed integer or floating point values



**Digital:** binary values 0 or 1



**String:** sequence of characters (maximum 80)



**Composite:** signed integer or floating point values generated as result of an operation between two gates



**Event/Alarm:** active depending on the value of the associated gates

To configure gates you just need to double click on the corresponding icon; it will then be possible to manage them with *Gate Builder*.

Channel	Device	Gate ID	N ID	Address	Description	Measure	Variable type	Tolerance	Min. value	Max. value
1	1	T1V	1	3.0002	Temperature T1 - Detected		DOUBLE		0	0
2	1	T2V	2	3.0003	Temperature T2 - Detected		DOUBLE		0	0
3	2	T2A	0	3.0005	Temperature T2 - Set		DOUBLE		0	0
4	2	T3V	1	3.0006	Temperature T3 - Detected		DOUBLE		0	0
5	3	T4V	0	3.0008	Temperature T4 - Detected		DOUBLE		0	0
6	3	T5V	0	3.0009	Temperature T5 - Detected		DOUBLE		0	0
7	4	T5A	0	3.0011	Temperature T5 - Set		DOUBLE		0	0
8	4	T6V	0	3.0012	Temperature T6 - Detected		DOUBLE		0	0
9	5	T7V	0	3.0013	Temperature T7 - Detected		DOUBLE		0	0
10	5	T7A	0	3.0014	Temperature T7 - Set		DOUBLE		0	0
11	6	T8V	0	3.0045	Temperature T8 - Detected		DOUBLE		0	0
12	6	T9V	0	3.0046	Temperature T9 - Detected		DOUBLE		0	0
13	6	T9A	0	3.0047	Temperature T9 - Set		DOUBLE		0	0
14	7	T10V	0	3.0026	Temperature T10 - Detected		DOUBLE		0	0
15	8	T10A	0	3.0028	Temperature T10 - Set		DOUBLE		0	0
16	9	T11V	0	3.0035	Temperature T11 - Detected		DOUBLE		0	0
17	9	T11A	0	3.0036	Temperature T11 - Set		DOUBLE		0	0
18	9	T12V	0	3.0037	Temperature T12 - Detected		DOUBLE		0	0
19	10	T12A	0	3.0055	Temperature T12 - Set		DOUBLE		0	0
20	11	T13V	0	3.0075	Temperature T13 - Detected		DOUBLE		0	0
21	12	T13A	0	3.0077	Temperature T13 - Set		DOUBLE		0	0
22	13	T14V	0	3.0054	Temperature T14 - Detected		DOUBLE		0	0
23	14	T14A	0	3.0090	Temperature T1 - Set		DOUBLE		0	0
24	14	T15V	0	3.0091	Temperature T15 - Detected		DOUBLE		0	0
25	15	T16V	0	3.0088	Temperature T16 - Detected		DOUBLE		0	0
26	15	T16A	0	3.0089	Temperature T16 - Set		DOUBLE		0	0

Open **Gate Builder** and select **Help** menu item for more details.

**Tip:**

Double Click+SHIFT opens the file using the program to which all the *Windows* files with .DBF extension (if existing) are related.

## 6 Code

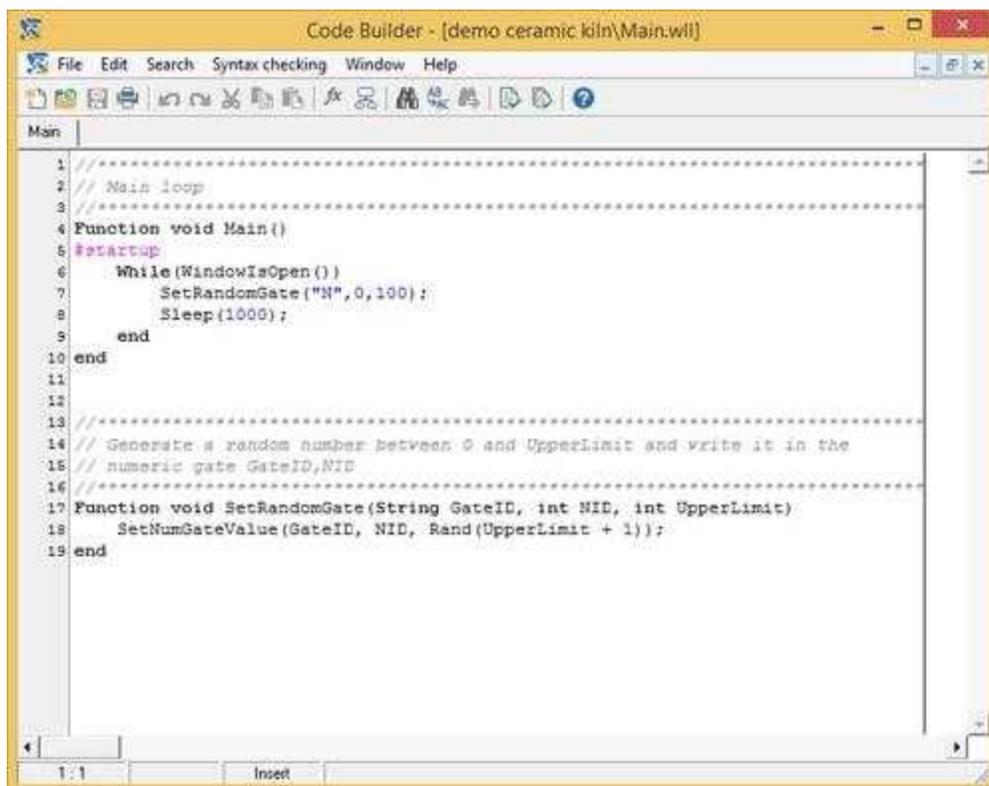
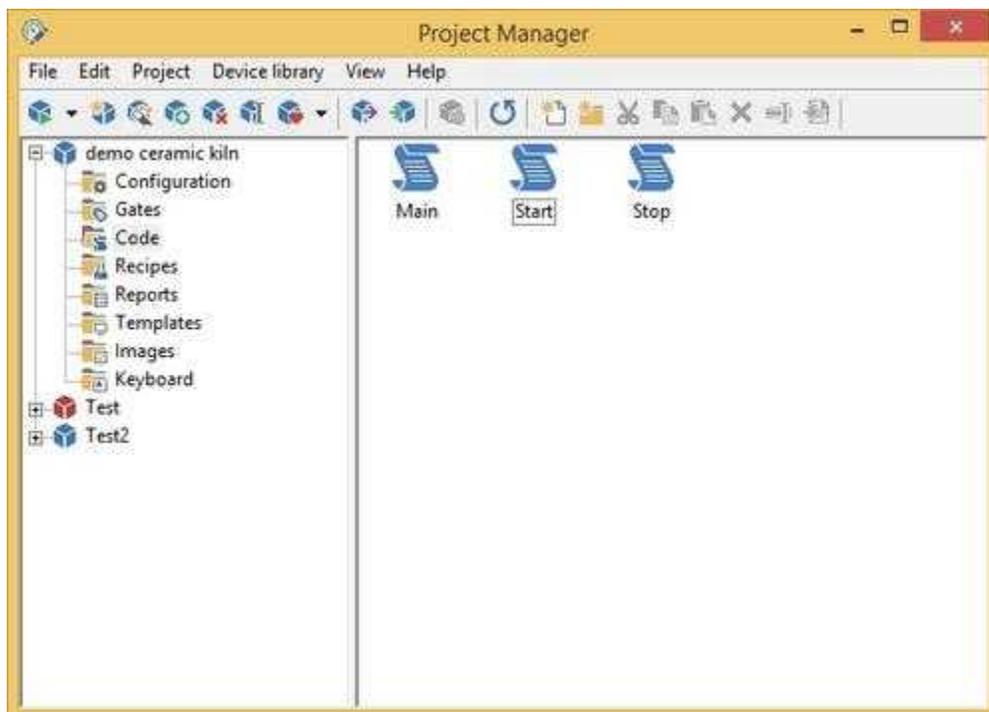
### 6.1 Code

Often, in a supervision project, comes the need to carry out special operations to the fulfillment of certain conditions or on operator request. Some examples include performing a calculation of a complex formula, send email or SMS, read or write text files, operate directly on gates value, manipulate certain objects of the synoptic displayed at that time, etc.

In this folder are grouped all the files containing the code that the software will have to execute

The creation of a new blank code file is performed by selecting the *Code* folder and then moving the mouse on the right window and pressing the right button: the "*New | File code*" menu entry is then displayed

*Code Builder* is started automatically by the *Project Manager* by double clicking on the icon to any page of language previously created.



**Tip:**

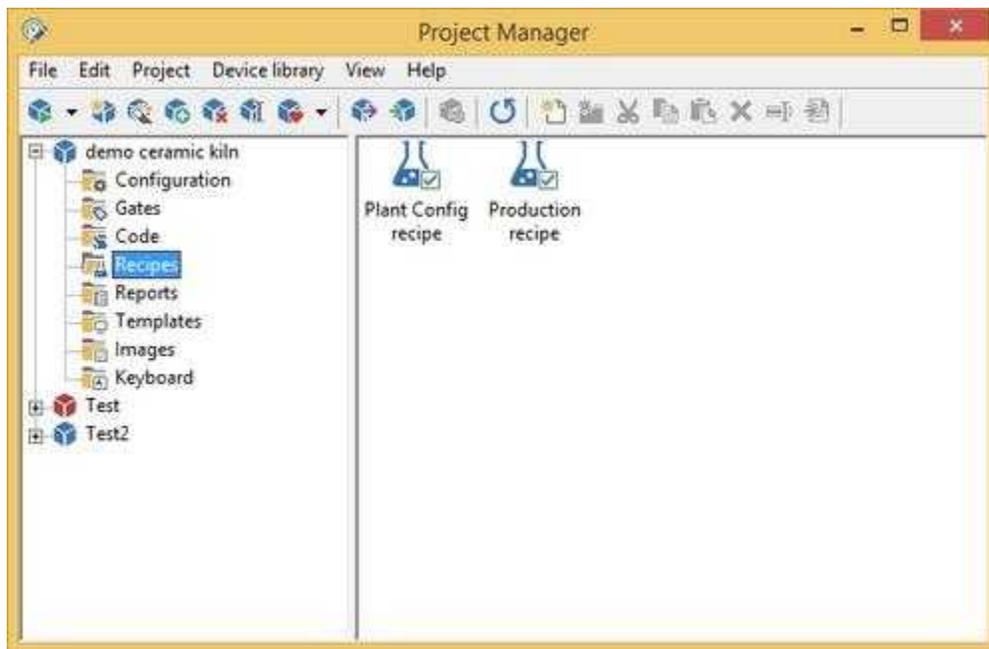
Double Click+SHIFT opens the file with the *Windows WordPad*

## 7 Recipes

### 7.1 Recipe models

A requirement that often meet in a supervision project is to be able to send to the process a group of parameters or configuration data in order to obtain a certain behavior of the process itself. Must also be possible to save this parameter group to a file and then use it again later when the need arises. Concrete examples of what is written can be a list of the set point temperature of the sections of a tunnel kiln for cooking ceramics: in relation to the set point in each section, can change the final characteristics of the product treated in the plant. Therefore it can be handy when you want to get the same product at different times, you can upload from an archive the specific settings for the processing and sending them quickly to the plant. All this can be managed through the *Recipes*: every recipe file contains within it all set points required to obtain a specific product. The list of variables or gates that make up the recipe are defined in the *Recipe model*. So first it must be defined a *Recipe model* from which then may be derived numerous recipes, each of which containing different set point and then related to different products.

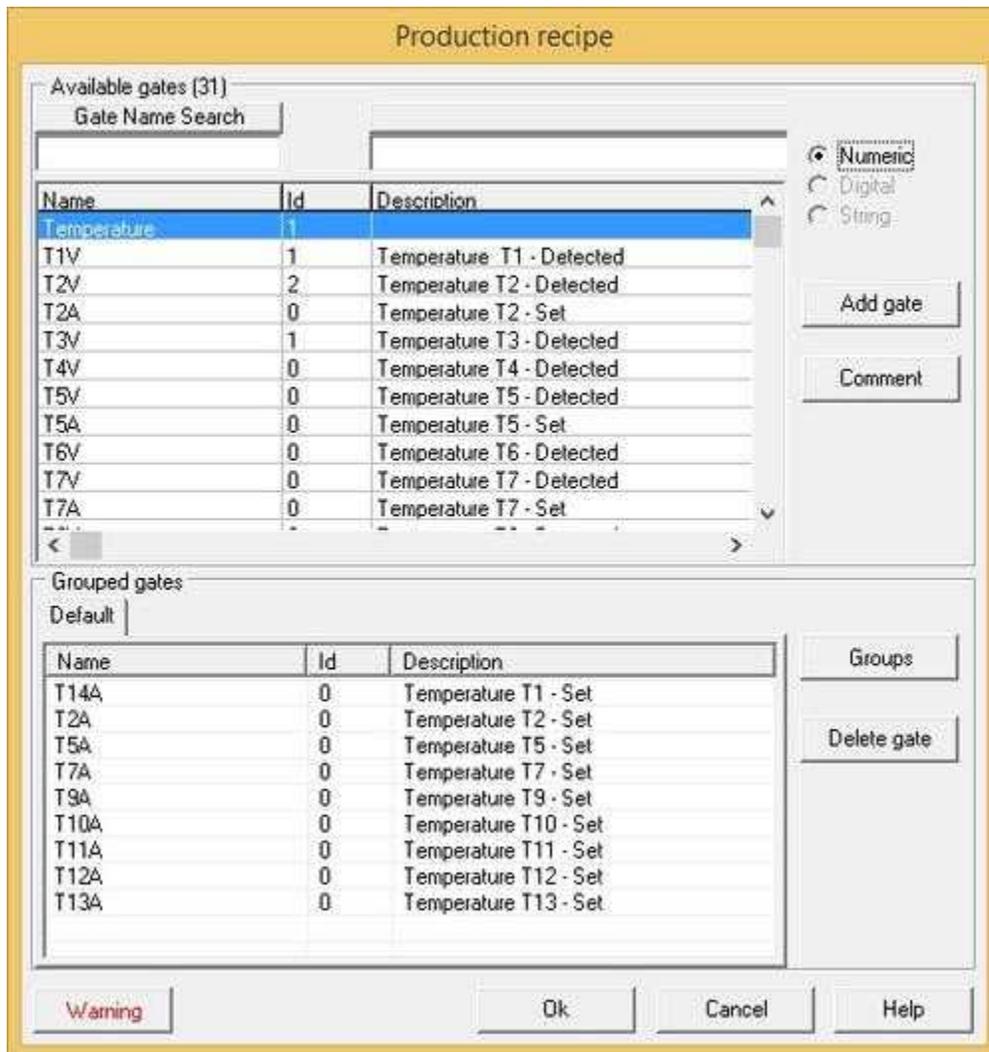
This folder contains all the defined recipe model.



By pressing the mouse right button on the right part of the screen, if the cursor has not selected any icon, it is possible to create a new model; on the contrary, if it has, you can execute the usual file operations (copy, cut, paste, rename).

To modify a model just double click on its icon.





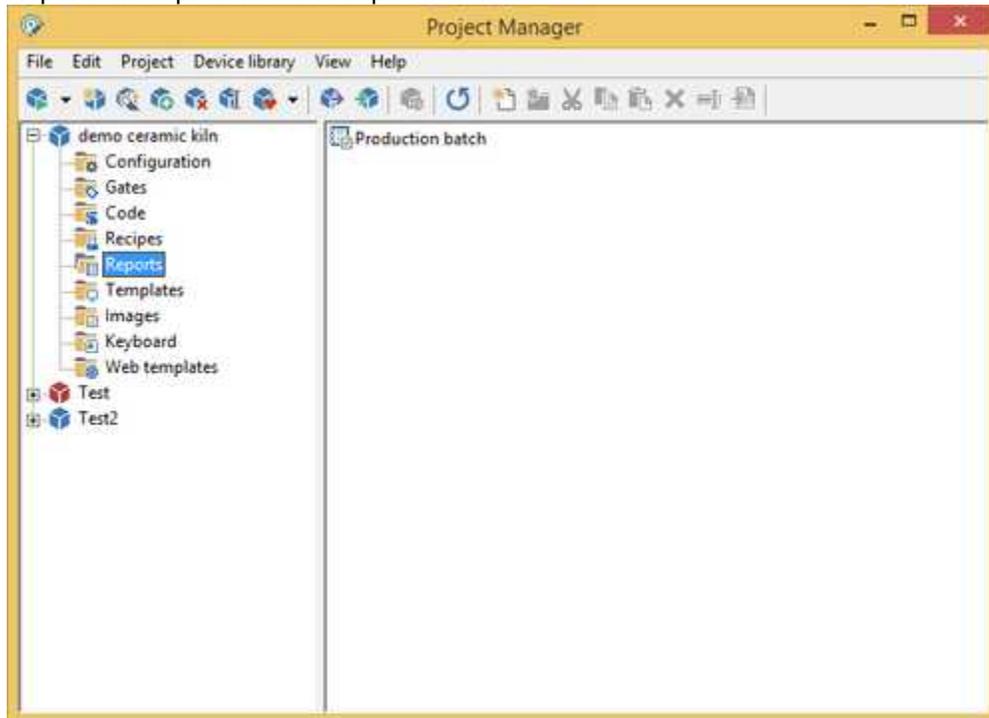
With the dialog box showed in figure it is possible to define the gate list of the model. In the upper part of the window it is possible to view the gates of the project, select and add it into the model. In the lower part there is the list of the gates of the model gathered in different groups. Using the *Delete gate* button, it is possible to delete the gates from the model. You can give a description to the model using the *Comment* button..

Press SHIFT + left mouse button to make a multiple selection of the gates.

## 8 Reports

### 8.1 Reports

Reports are representations of process data.

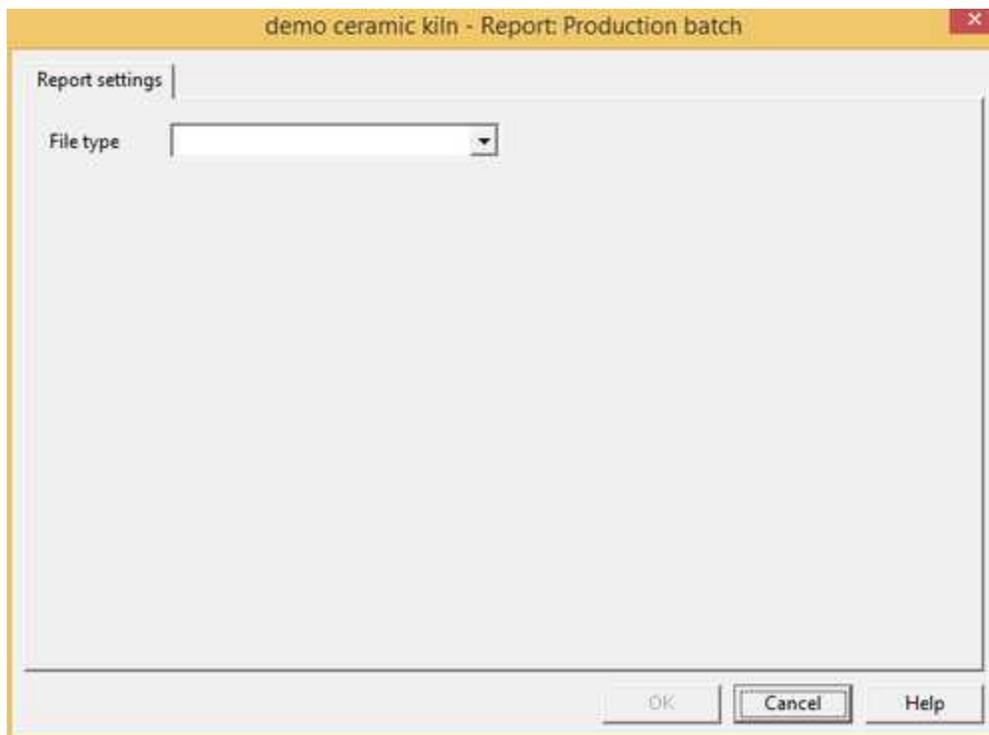


This folder contains all defined report models. By pressing the mouse right button on the right part of the screen, if the cursor has not selected any icon, it is possible to create a new report model; on the contrary, if it has, you can execute the usual file operations (copy, cut, paste, rename).

To modify a report model, you just need to double click on the corresponding icon.



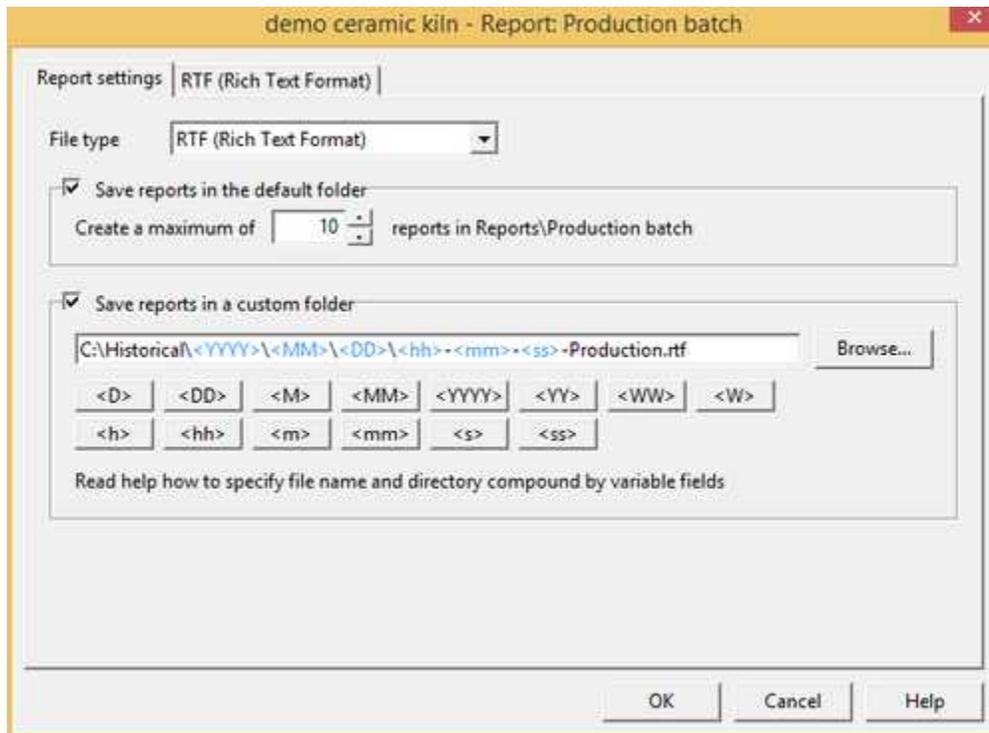
Creating a new report model:



First, choose a file format for the report. Select it using *File type* combo box:

- *TXT (Standard Text File)*: it can contain data only in simple text format
- *RTF (Rich Text Format)*: it can contain data in text format (with font and color definition) and images. RTF reports may be converted into PDF (Portable Document Format); PDF allows smaller files and security (you can set a password to prevent modifying)
- *DAT (Formatted Text)*: it contains data records saved in formatted text compatible with CSV (Comma Separated Values) file format; each record is a list of variables (gates).

After you choose report format, you will be able to set other configuration parameters.



*Save files in <project folder>\Reports<Report model folder>:* enable/disable creation of reports in the showed default directory

*Create a maximum of...:* number of reports to keep on disk. Each generated report has the same filename as the name of the report model, but the extension is an index between 001 and the value inserted in this field. The report file that has 001 extension is the last created report (the more recent). Report files are generated and saved in a sub directory of the project folder (it is Reports\<Report name>); its name is showed at the right of the edit box.

*Save files in custom directory:* enable/disable creation of reports in the specified custom directory; if this flag is enabled, a full path file name must be specified. If the path does not exist, it will be automatically created in the runtime during report generation. Some special strings can be specified in the path file name: when a new report file will be generated (in the runtime) they will be substitute with the values that they represent. Following there is the list of possible special strings:

Special strings	Description	Example
<D>	current day in short format	(1="1",...,25="25")
<DD>	current day on 2 chars	(1="01",...,25="25")
<M>	current month in short format	(1="1",...,12="12")
<MM>	current month on 2 chars	(1="01",...,12="12")
<YYYY>	current year on 4 chars	(2011="2011",...,2015="2015")
<YY>	current year last 2 significant digits	(2011="11",...,2015="15")
<h>	current hour in short format	(1="1",...,15="15")
<hh>	current hour on 2 chars	(1="01",...,15="15")
<m>	current minute in short format	(1="1",...,15="15")
<mm>	current minute on 2 chars	(1="01",...,15="15")
<s>	current second in short format	(1="1",...,15="15")
<ss>	current second on 2 chars	(1="01",...,15="15")

*Example 1:* suppose to specify the following path file name for a RTF report.

**C:\Historical\<YYYY>\<MM>\<DD>\<hh>-<mm>-<ss>-Production.rtf**

Suppose that Runtime generates this report in date 17 June 2011 at 10:47:50: the following path will be automatically created during report creation: **C:\Historical\2011\06\17\** and the report file name will be **10-47-50-Production.rtf**

*Example 2:* suppose to specify the following path file name for a DAT report.

**C:\Historical\<YYYY>\<MM>\<DD>\<YY>-<MM>-<DD>-Production.dat**

Suppose to start Runtime in date 17 June 2011 : the following path will be automatically created during report creation: **C:\Historical\2011\06\17\** and the report file name will be **2011-06-17-Production.dat**. All the records will be added to this file until a new day; when the day change, the following new path will be automatically created **C:\Historical\2011\06\18\** and the following new report file will be produced: **2011-06-18-Production.dat**

**Note:** reports saved in the custom directory will be not automatically deleted when the free disk space is lower than the specified threshold.

## 8.2 TXT reports

**TXT (Standard Text File)** reports can contain data only in simple text format. This a sample of TXT report.

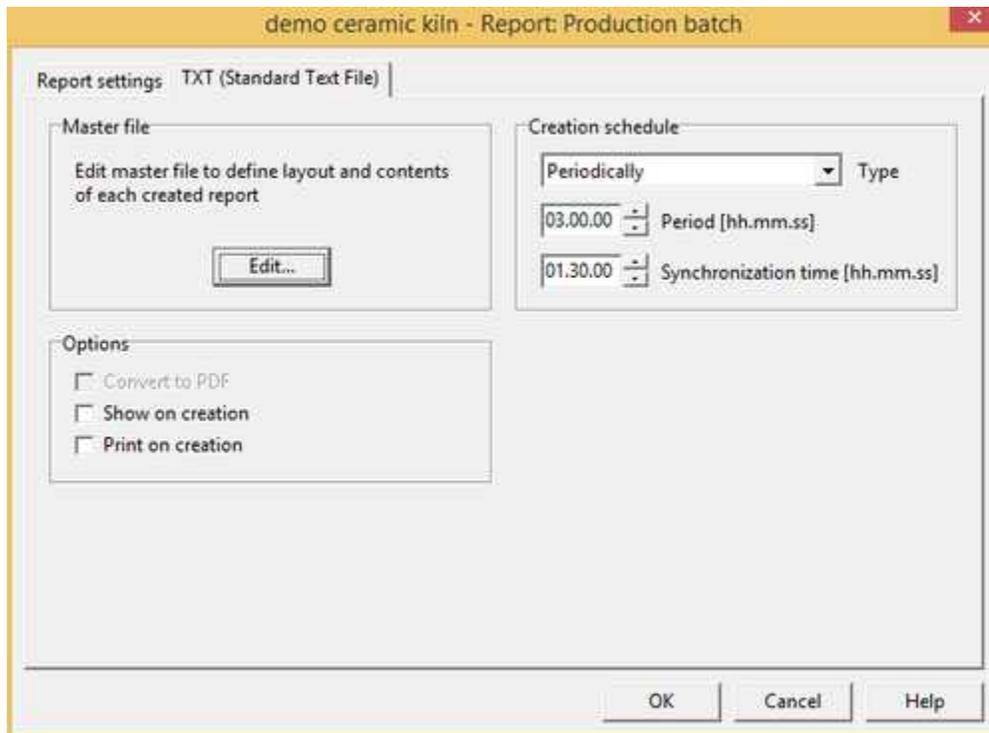
```

***** PRODUCTION REPORT *****
Date: 03/11/2008      Time: 13.45.34

Product code ..... FX145-I
Format .....(cm²) 100
Cycle time .....(min) 54
Weight .....(g) 20
Average speed .....(m/min) 1.12
Batch time .....(min) 145
Total production .....(kg) 3
Made units ..... 142
Production waste .....(%) 5
    
```

TXT reports are different from DAT reports. TXT reports are created and "closed", they can not be modified in the future.

### **Report settings page options**



### **Master file**

Report master file defines layout and content of each created report. Master file can contain fixed text and commands. Commands allow to insert variable strings as gates value or results of data computation. Commands must be surrounded by curly brackets ( { and } ) and can be a single code instruction (API) or user-defined function defined in a code file of the project (use *CodeBuilder* to edit code files and to create user-defined functions). Refer to inline help of *CodeBuilder* (API > Report section) to get more information.

To create or edit the master file press *Edit* button. The system application *WordPad* will be run; modify the file and save it.

Below there is a sample master file; it may be the master file of the above sample report. Used commands call same API (*GetDateString()*, *GetTimeString()*, *GetStrGateValue()* and *GetNumGateValue()*) and a user-defined function (*GetProdWaste()*); the first row is a "comment", thus it will be discarded during report generation.

```

{ // Master file report }

***** PRODUCTION REPORT *****

Date: {GetDateString("/",true)}   Time: {GetTimeString(":")}

Product code ..... {GetStrGateValue("Code", 0)}
Format .....(cm2) {GetNumGateValue("Format", 0)}
Cycle time .....(min) {GetNumGateValue("CycleTime", 0)}
Weight .....(g) {GetNumGateValue("Weight", 0)}
Average speed .....(m/min) {GetNumGateValue("MeanSpeed", 0)}
Batch time .....(min) {GetNumGateValue("BatchTime", 0)}
Total production .....(kg) {GetNumGateValue("Production", 0)}
Made units ..... {GetNumGateValue("Units", 0)}
Production waste.....(%) {GetProdWaste(GetStrGateValue("Code", 0))}

```

### **Creation schedule**

Using controls of this group you can define when a report is generated. A report may be created automatically at specified frequency or may be created manually. You can create a report manually selecting menu item *Report > Create* in *Runtime* or by code calling the instruction *ReportCreate()*. *Type* defines how often reports will be created:

- *Manually*: auto generation is disabled. To create a report manually choose menu item *Reports*

- > *Create in Runtime* or by code call the API `ReportCreate()`.
- *Periodically*: reports are created periodically; insert creation period in the *Creation period* field (for example: if you insert `03:00:00`, will be created a report every 3 hours). *Synchronization time* defines the time from which the periodicity counter starts (if you insert `01:30:00`, reports are created at 01:30, 04:30, 07:30, ...)
- *Weekly*: reports will be created once a week on the day specified by *Day* field and at the time specified by *Synchronization time* field.
- *Monthly*: reports will be created once a month on the day specified by *Day* field and at the time specified by *Synchronization time* field.
- *Yearly*: reports will be created once a year on the day specified by *Day* and *Month* fields and at the time specified by *Synchronization time* field.

**Options**

*Show on creation*: if you flag this box, when a report is created, it will be showed using the system application *WordPad*.

*Print on creation*: if you flag this box, when a report is created, it will be printed using the system application *WordPad*. Used printer and page layout are the ones defined in such program.

Check box *Convert to PDF format* is disabled because TXT reports can not be converted to PDF format; use RTF reports instead.

### 8.3 RTF reports

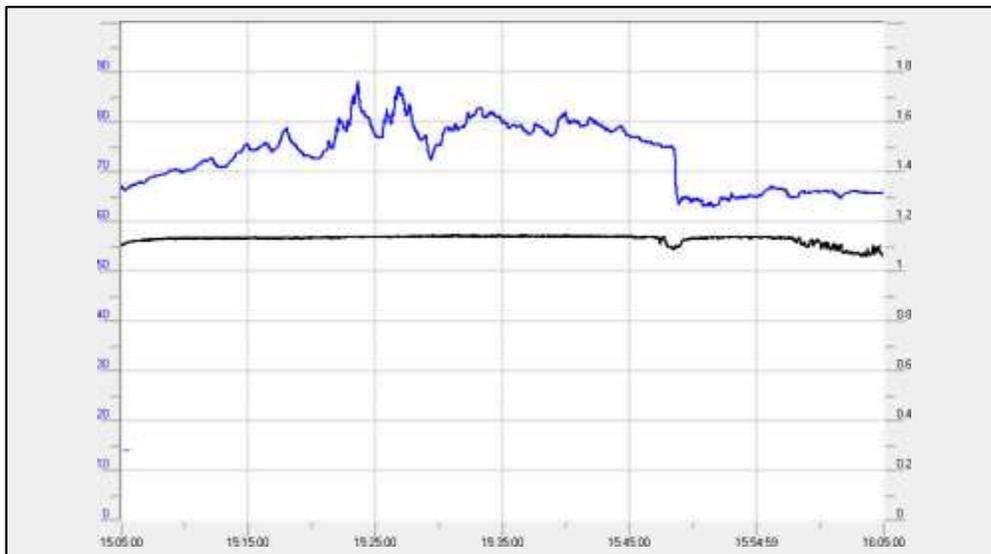
**RTF (Rich Text Format)** reports can contain data in text format (with font and color definition) and images. RTF reports may be converted into PDF (Portable Document Format); PDF allows smaller files and security (you can set a password to prevent modifying).

This a sample of RTF report.

***PRODUCTION REPORT***

Date: 03/11/2008  
Time: 13.45.34

Product code .....	FX145-I
Format .....	(cm <sup>2</sup> ) 100
Cycle time .....	(min) 54
Weight .....	(g) 20
Average speed .....	(m/min) 1.12
	)
Batch time .....	(min) 145
Total production .....	(kg) 3
Made units .....	142
Production waste .....	(%) 5



RTF reports are different from DAT reports. RTF reports are created and "closed", they can not be modified in the future.

### Report settings page options

### **Master file**

Report master file defines layout and content of each created report. Master file can contain fixed text and commands. Commands allow to insert variable strings as gates value or results of data computation. Commands must be surrounded by curly brackets ( { and } ) and can be a single code instruction (API) or user-defined function defined in a code file of the project (use *CodeBuilder* to edit code files and to create user-defined functions). Refer to inline help of *CodeBuilder* (API > Report section) to get more information.

To create or edit the master file press *Edit* button. The system application *WordPad* will be run; modify the file and save it.

Below there is a sample master file; it may be the master file of the above sample report. Used commands call same API (`GetDateString()`, `GetTimeString()`, `GetStrGateValue()`, `GetNumGateValue()` and `ReportInsertTemplateEx()`) and a user-defined function (`GetProdWaste()`); the first row is a "comment", thus it will be discarded during report generation.

```

{// Master file sample }
PRODUCTION REPORT
Date:      {GetDateString("/",true)}
Time:      {GetTimeString(":")}

Product code ..... {GetStrGateValue("Code", 0)}
Format ..... (cm²){GetNumGateValue("Format", 0)}
Cycle time ..... (min){GetNumGateValue("CycleTime", 0)}
Weight ..... (g){GetNumGateValue("Weight", 0)}
Average speed ..... (m/min){GetNumGateValue("MeanSpeed", 0)}
Batch time ..... (min){GetNumGateValue("BatchTime", 0)}
Total production ..... (kg){GetNumGateValue("Production", 0)}
Made units ..... {GetNumGateValue("Units", 0)}
Production waste ..... (%) {GetProdWaste(GetStrGateValue("Code", 0))}

{ReportInsertTemplateEx("ReportChart")}
    
```

**Creation schedule**

Using controls of this group you can define when a report is generated. A report may be created automatically at specified frequency or may be created manually. You can create a report manually selecting menu item *Report > Create* in *Runtime* or by code calling the instruction `ReportCreate()`.

*Type* defines how often reports will be created:

- *Manually*: auto generation is disabled. To create a report manually choose menu item *Reports > Create* in *Runtime* or by code call the API `ReportCreate()`.
- *Periodically*: reports are created periodically; insert creation period in the *Creation period* field (for example: if you insert *03:00:00*, will be created a report every 3 hours). *Synchronization time* defines the time from which the periodicity counter starts (if you insert *01:30:00*, reports are created at 01:30, 04:30, 07:30, ...)
- *Weekly*: reports will be created once a week on the day specified by *Day* field and at the time specified by *Synchronization time* field.
- *Monthly*: reports will be created once a month on the day specified by *Day* field and at the time specified by *Synchronization time* field.
- *Yearly*: reports will be created once a year on the day specified by *Day* and *Month* fields and at the time specified by *Synchronization time* field.

**Options**

*Convert to PDF format*: select this option to automatically convert reports in PDF (Portable Document Format). PDF allows smaller files, a better control of page dimensions and security issues (you can set a password to prevent modifying them). If you mark this check box a new tab sheet will appear; in this tab sheet you can define PDF conversion settings; refer to instructions below to get more information about those parameters.

*Show on creation*: if you flag this box, when a report is created, automatically it will be showed. RTF reports are showed using the system application *WordPad*. PDF reports are showed using the application that is associated in the system to PDF documents; so you need to install separately a program able to open and read PDF files (*Adobe Reader*, *Foxit Reader*, ...).

*Print on creation*: if you flag this box, when a report is created, automatically it will be printed. RTF reports are printed using the system application *WordPad*, so used printer and page layout are the ones defined in such program. PDF reports are printed using the application that is associated in the system to PDF documents, so the used printer is the one defined in such program.

**PDF format conversion settings page options**

**Measure unit:** select the measure unit (*Centimeters* or *Inches*) in which you want to express all parameters of this page.

#### **Sheet size**

**Format:** is the PDF output sheet size; choose one of the standard sizes or *Custom* to specify a particular size.

**Width:** for *Custom* format, is the sheet width.

**Height:** for *Custom* format, is the sheet height.

#### **Orientation**

Choose sheet orientation (*Vertical* or *Horizontal*).

#### **Page margins**

Insert extent of *Left*, *Right*, *Top* and *Bottom* margins. Margins defines the print area of the sheet; original RTF content will be fitted in this area.

#### **Border**

It is possible to add a decorative border. Border may be drawn out the print area defined by *Page margins*.

**Type:** is the type of the border; choose *None* if you do not want to add a border.

**Width:** is the border width.

**Distance:** is the distance between the border and the edge of the sheet.

**Color:** border color.

#### **Header / Footer**

A line of text can be automatically added on each page of the report as header or footer. The content of the text can be made up of fixed text and special fields like *<Page>* and *<Name>*.

When generating the report, the *<Page>* field will be automatically replaced with the page numbering in the following format: 1/4, 2/4, 3/4, 4/4 ...

The *<Name>* field instead will be replaced with the name associated with the report.

#### **Security restrictions**

One of the advantages of the PDF format is the possibility to add security restrictions to created reports.

*Require a password to open the document:* flag this option to require to users a password to open and read the created reports. Password must be specified in *Password* field.

*Allow changing of document content:* if you check this box, PDF reports can be edited using a PDF-authoring application. Only password protected reports may be protected against editing.

*Allow copying/extraction of document content:* if you check this box, PDF reports content can be copied/extracted to be pasted in other documents. Only password protected reports may be protected against content copying/extraction.

## 8.4 DAT reports

**DAT (Formatted Text)** reports are files containing records of variables (gates) in formatted text. DAT reports are compatible with CSV (Comma Separated Values) format.

This is a sample of DAT report.

dd/MM/yyyy	H.mm.ss	T1V0	T2V0	T3V0	T4V0	T5V0
17/01/2005	21.51.19	67,00	1.004,00	1050,00	1.100,00	1.219,00
17/01/2005	21.51.29	67,00	999,00	1050,00	1.100,00	1.214,00
17/01/2005	21.51.39	67,00	994,00	1050,00	1.100,00	1.209,00
17/01/2005	21.51.49	67,00	989,00	1050,00	1.100,00	1.204,00
17/01/2005	21.51.59	67,00	994,00	1050,00	1.100,00	1.199,00
17/01/2005	21.52.09	67,00	999,00	1050,00	1.100,00	1.194,00
17/01/2005	21.52.19	67,00	1.004,00	1050,00	1.100,00	1.189,00
17/01/2005	21.52.29	67,00	1.009,00	1050,00	1.100,00	1.184,00
17/01/2005	21.52.39	67,00	1.008,00	1050,00	1.100,00	1.179,00
17/01/2005	21.52.49	67,00	1.003,00	1050,00	1.100,00	1.184,00
17/01/2005	21.52.59	67,00	998,00	1050,00	1.100,00	1.189,00
17/01/2005	21.53.09	67,00	993,00	1050,00	1.100,00	1.194,00
17/01/2005	21.53.19	67,00	990,00	1050,00	1.100,00	1.199,00
17/01/2005	21.53.29	67,00	995,00	1050,00	1.100,00	1.204,00

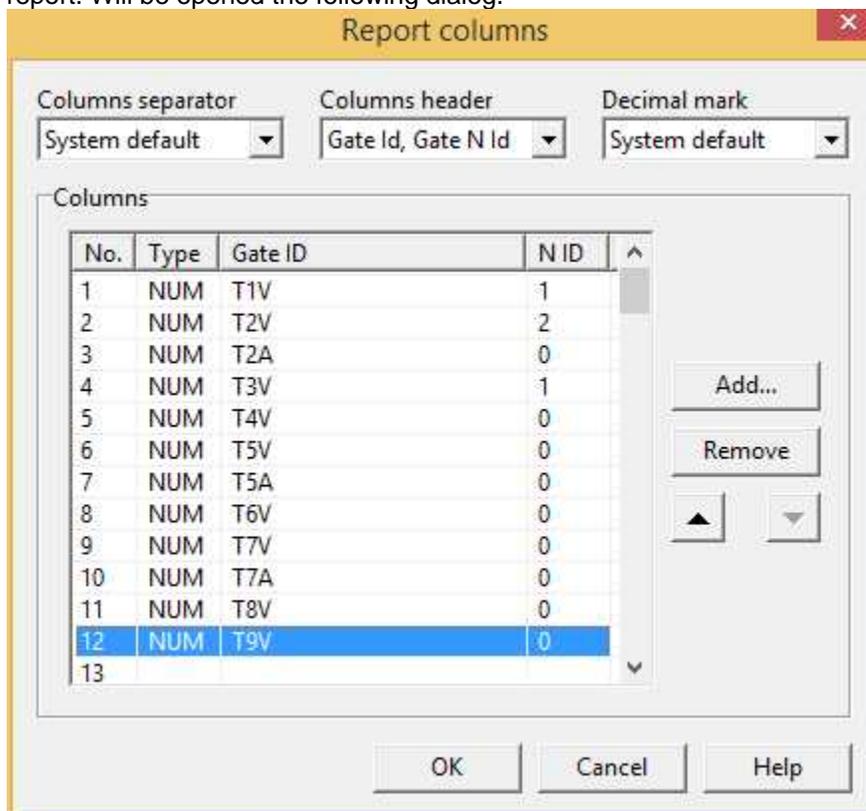
DAT reports are different from TXT and RTF reports. DAT reports are not created all in one time; they are continuously updated appending new records at fixed frequency or by code instruction. Each record is composed of columns: the first two columns are record date and record time, other columns are values of the gates selected to record in the report.

### **Report settings page options**



### **Columns**

Clicking on *Edit* button it is possible to define which gates have to be saved in each record of the DAT report. Will be opened the following dialog.



*Columns separator*: select the separator character between columns of the records.

*Columns header:* choose columns header; it may be the string obtained concatenating gate ID and gate N ID or it may be the gate description.

*Decimal mark:* select the mark for double numbers.

*Columns:* using *Add*, *Remove*, ▲ and ▼ buttons, define which gates have to compose each record and their order. Numeric, digital, compound and string gates are saved as the value assumed at the record storing time. Event/alarm gates are saved as a string; the string is the event/alarm message if the event/alarm is active at the record storing time; otherwise the saved string is blank. It is possible to select up to 99 gates.

**Report appending mode**

You can choose to append record automatically or manually. Choose "*Automatically, at fixed frequency*" if you want to append records at fixed frequency; appending period must be defined. Choose "*Manually, using code functions*", if you want to append record by code instructions; in this case every time you want to append a new record you need to call the API `ReportAppendRecord()`.

**Enable for record appending**

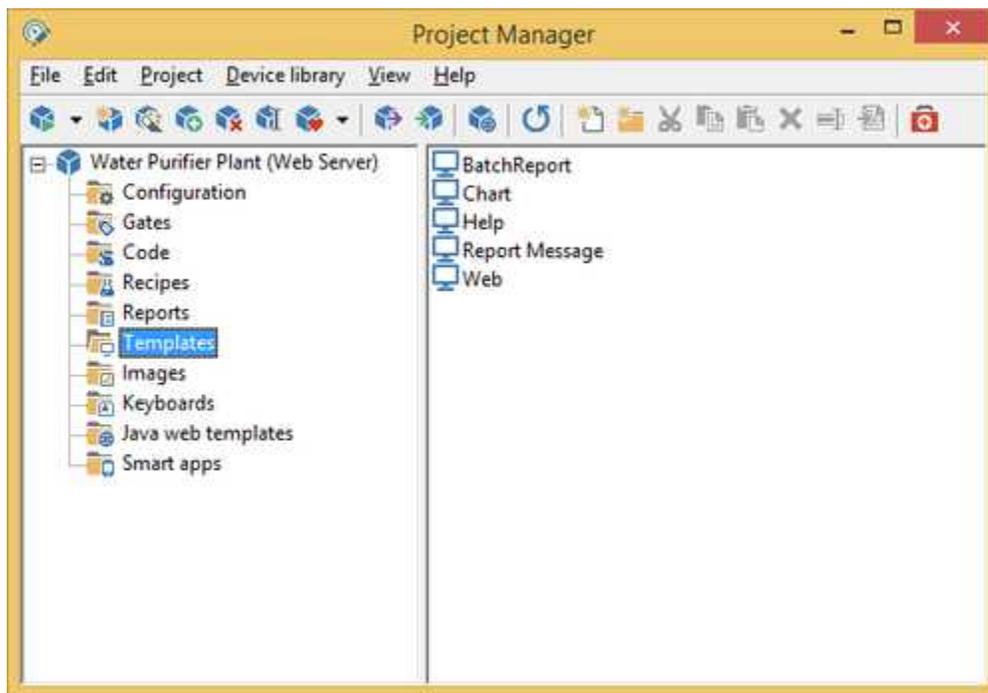
You can choose to append record always or only when a trigger is verified. Choose "*Always*" if you want to append records for all the period that the Runtime is active. Choose "*On trigger*", if you want to append only when the trigger condition is verified.

**Trigger for record appending**

If you choose automatically appending mode you have to specify when recording is enabled. Choose a gate and a condition; when condition applied to gate changes from false to true (the trigger becomes active) a new report is created. When trigger is active records are appended at the frequency specified. When trigger becomes inactive (condition on the selected gates changes form true to false) recording stops. Next trigger activation forces the creation of a new report file.

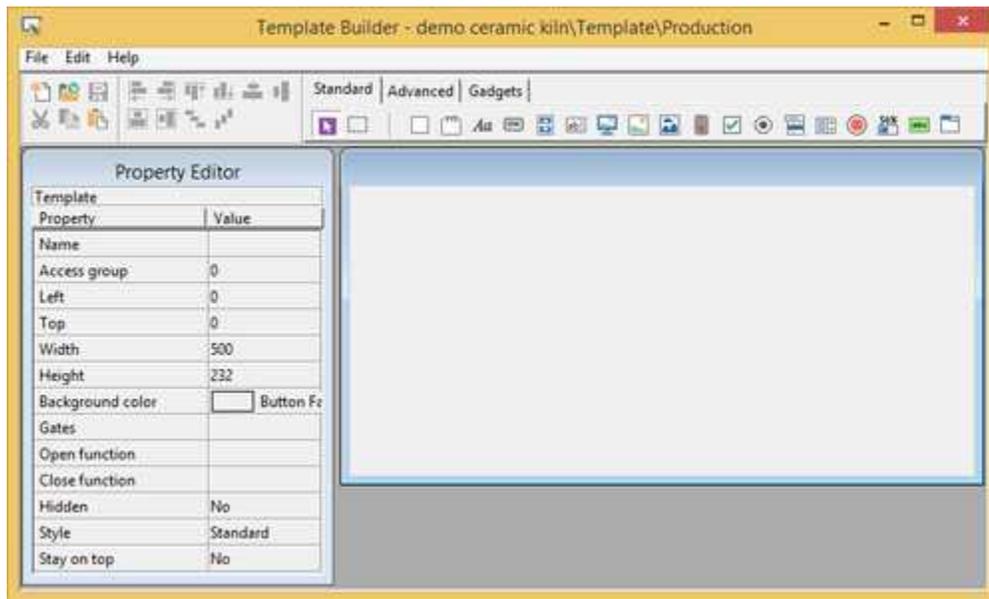
## 9 Template

### 9.1 Template



In this folder there are all the supervision pages of the server application. By pressing the mouse right button on the right part of the screen, if the cursor has not selected any icon, it is possible to create a new Template; on the contrary, if it has, you can execute the usual file operations (copy, cut, paste, rename).

To access a file you just need to double click on the corresponding icon; it will then be possible to build the page with *Template Builder*.



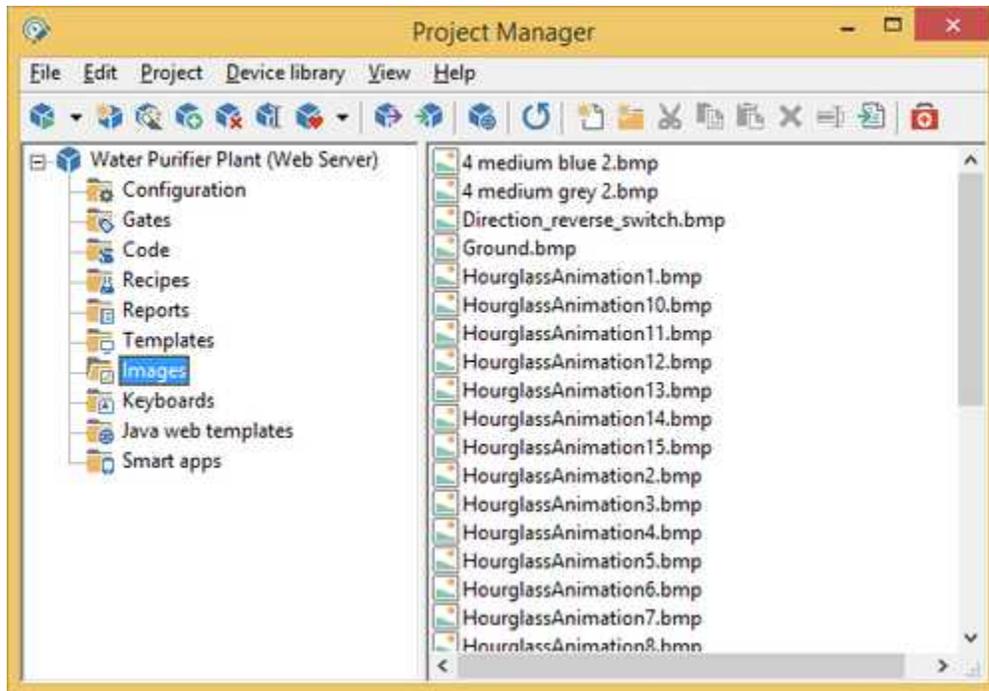
Open **Template Builder** and select **Help** menu item for more details.

**Tip:**

Double Click+SHIFT opens the file with the *Windows WordPad*

## 10 Images

### 10.1 Images



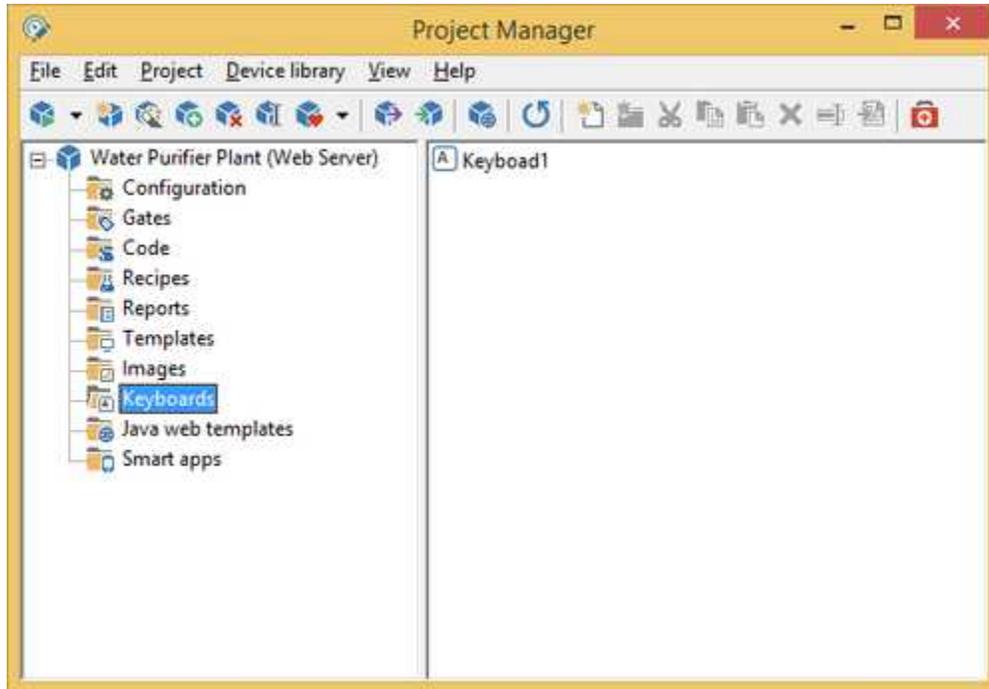
In this folder there are all the images concerning the project. By pressing the mouse right button, on the right part of the screen, if the cursor has not selected any icon, it is possible to create a new file; on the contrary, if it has, you can execute the usual file operations (copy, cut, paste, rename).

**Tip:**

Double Click opens the image using the program to which the *Windows* images files (if existing) are related.

## 11 Keyboards

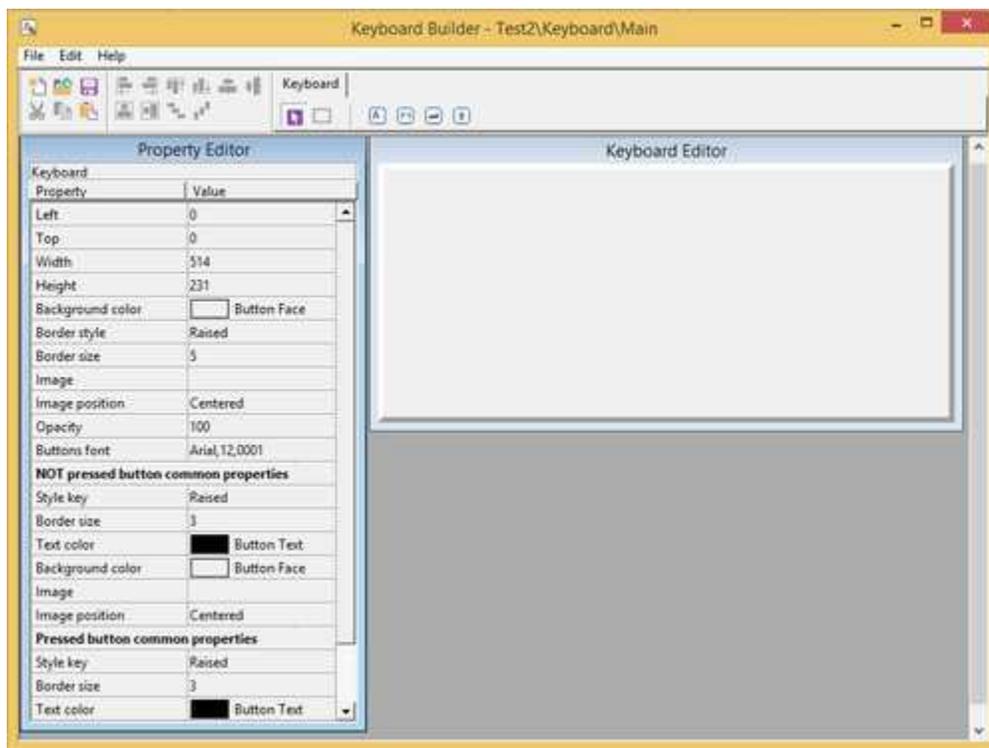
### 11.1 Keyboards



When the application is executed on a computer with no hardware keyboard and with touch screen monitor, there is the need to use virtual keyboard for data input.

In this folder there are all the virtual keyboards defined for the application. By pressing the mouse right button on the right part of the screen, if the cursor has not selected any icon, it is possible to create a new virtual keyboard; on the contrary, if it has, you can execute the usual file operations (copy, cut, paste, rename).

To access a file you just need to double click on the corresponding icon; it will then be possible to build the keyboard with *Keyboard Builder*.



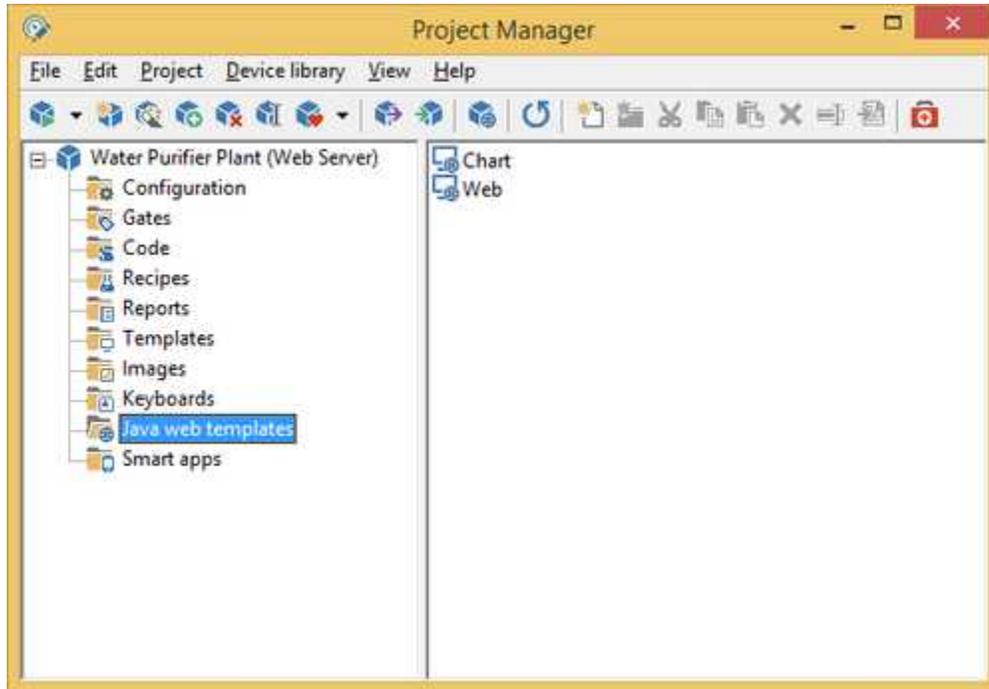
Open **Keyboard Builder** and select **Help** menu item for more details.

**Tip:**

Double Click+SHIFT opens the file with the *Windows WordPad*

## 12 Java web templates

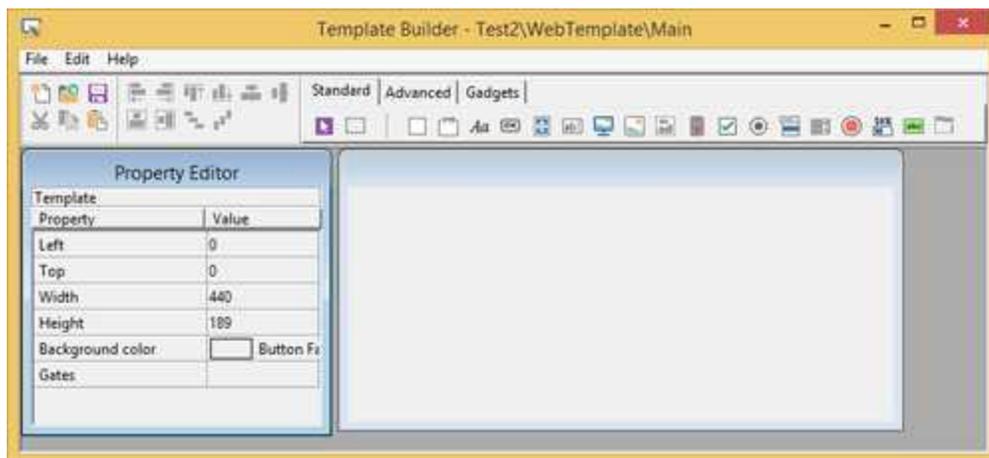
### 12.1 Java web templates



This folder contains all the synoptic that will be used to build the Java web client application.

To access a file you just need to double click on the corresponding icon; it will then be possible to build the page with *Template Builder*.

Java web templates pages support a subset of objects that are supported by Template pages: unsupported objects will be automatically disabled and removed from *Template Builder* during page loading and saving operations.



## 13 Smart apps

### 13.1 Smart apps

Often there is the need to monitor and control the process from Smartphone or Tablet. In this folder you can quickly create a Smart app able to interact with the server application to monitor and modify variables that are part of the supervision process.

*Smart App Builder* is automatically started from *Project Manager* by double clicking the icon labeled "Default" in the project *Smart apps* section.

