

The Siemens logo is displayed in a white rectangular box in the top right corner of the image. The background of the entire page is a blurred industrial factory setting with a man in a light blue shirt using a tablet. Overlaid on the image are various digital icons: a '24/7' circular arrow icon, a 'NEWS' icon with a person silhouette, a 'Home' icon, and a network diagram with three people icons and binary code. The text 'Industry Online Support' is also overlaid in a stylized font.

SIEMENS

Continuous Integration with TIA Portal: Setup

TIA Portal / Continuous Integration

<https://support.industry.siemens.com/cs/ww/en/view/109801203>

Siemens
Industry
Online
Support



Legal information

Use of application examples

Application examples illustrate the solution of automation tasks through an interaction of several components in the form of text, graphics and/or software modules. The application examples are a free service by Siemens AG and/or a subsidiary of Siemens AG ("Siemens"). They are non-binding and make no claim to completeness or functionality regarding configuration and equipment. The application examples merely offer help with typical tasks; they do not constitute customer-specific solutions. You yourself are responsible for the proper and safe operation of the products in accordance with applicable regulations and must also check the function of the respective application example and customize it for your system.

Siemens grants you the non-exclusive, non-sublicensable and non-transferable right to have the application examples used by technically trained personnel. Any change to the application examples is your responsibility. Sharing the application examples with third parties or copying the application examples or excerpts thereof is permitted only in combination with your own products. The application examples are not required to undergo the customary tests and quality inspections of a chargeable product; they may have functional and performance defects as well as errors. It is your responsibility to use them in such a manner that any malfunctions that may occur do not result in property damage or injury to persons.

Disclaimer of liability

Siemens shall not assume any liability, for any legal reason whatsoever, including, without limitation, liability for the usability, availability, completeness and freedom from defects of the application examples as well as for related information, configuration and performance data and any damage caused thereby. This shall not apply in cases of mandatory liability, for example under the German Product Liability Act, or in cases of intent, gross negligence, or culpable loss of life, bodily injury or damage to health, non-compliance with a guarantee, fraudulent non-disclosure of a defect, or culpable breach of material contractual obligations. Claims for damages arising from a breach of material contractual obligations shall however be limited to the foreseeable damage typical of the type of agreement, unless liability arises from intent or gross negligence or is based on loss of life, bodily injury or damage to health. The foregoing provisions do not imply any change in the burden of proof to your detriment. You shall indemnify Siemens against existing or future claims of third parties in this connection except where Siemens is mandatorily liable.

By using the application examples you acknowledge that Siemens cannot be held liable for any damage beyond the liability provisions described.

Other information

Siemens reserves the right to make changes to the application examples at any time without notice. In case of discrepancies between the suggestions in the application examples and other Siemens publications such as catalogs, the content of the other documentation shall have precedence.

The Siemens terms of use (<https://support.industry.siemens.com>) shall also apply.

Security information

Siemens provides products and solutions with industrial security functions that support the secure operation of plants, systems, machines and networks.

In order to protect plants, systems, machines and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial security concept. Siemens' products and solutions constitute one element of such a concept.

Customers are responsible for preventing unauthorized access to their plants, systems, machines and networks. Such systems, machines and components should only be connected to an enterprise network or the internet if and to the extent such a connection is necessary and only when appropriate security measures (e.g. firewalls and/or network segmentation) are in place.

For additional information on industrial security measures that may be implemented, please visit

<https://www.siemens.com/industrialsecurity>.

Siemens' products and solutions undergo continuous development to make them more secure. Siemens strongly recommends that product updates are applied as soon as they are available and that the latest product versions are used. Use of product versions that are no longer supported, and failure to apply the latest updates may increase customer's exposure to cyber threats.

To stay informed about product updates, subscribe to the Siemens Industrial Security RSS Feed under

<https://www.siemens.com/cert>.

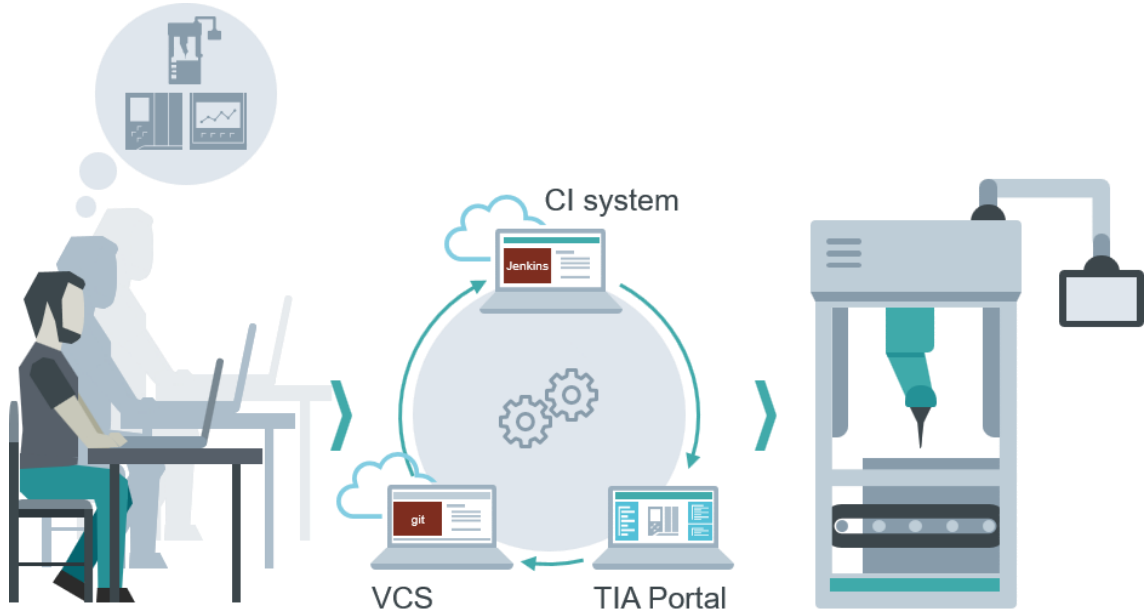
Table of contents

Legal information	2
1 Introduction	4
1.1 Overview	4
1.2 Variants	5
1.3 Principle of operation	6
1.4 Sequence	7
1.5 Components used	8
2 Installations	9
2.1 Installing the Continuous Integration server	9
2.1.1 Software	9
2.1.2 Setting up Jenkins	9
2.2 Version control server installation	9
2.2.1 Software	9
2.2.2 Setting up the project server	10
2.2.3 Project Server Listen Console setup	10
2.3 Automation desktop installation	11
2.3.1 Software	11
2.3.2 SIMATIC PLCSIM Advanced setup	11
2.3.3 Setting up Project Check for TIA Portal	11
2.3.4 Setting up Project Server Export Console	12
2.3.5 Setting up Transforming XML	12
2.3.6 Execution environment for Jenkins	12
2.3.7 Jenkins agent	14
2.4 Engineering desktop installation	15
2.4.1 Software	15
2.4.2 TIA Portal setup	15
3 Appendix	16
3.1 Service and support	16
3.2 Industry Mall	17
3.3 Links and literature	17
3.4 Change documentation	17

1 Introduction

1.1 Overview

Figure 1-1: Continuous Integration



Agility in automation projects

- Regular recording of changes to program blocks in a version management system (TIA Portal project server, version control system, e.g. with git)
- Automated building and testing in a "Continuous Integration System", for example with Jenkins
- Feedback to all stakeholders

Benefits

- Early detection of integration problems
- User programs written in short cycles
- Test changes automatically, regularly, and in less time

Scope

This application example demonstrates a simple workflow for implementation of parts of the Continuous Integration process with the Siemens portfolio.

It offers a modular toolset as an optimized basis for customer-specific solutions.

Scenario

A typical scenario begins when the user changes something in his/her automation program with TIA Portal and saves the changes in that user's version control system, e.g. the project server. These changes are automatically detected and the information is sent to the Continuous Integration system – in this case, the orchestration software Jenkins. Jenkins starts the test environment and performs the tests, for example to check against a programming style guide. The result of the tests is sent to Jenkins, adjusted for ease of analysis, and then the results are automatically sent to the developers so that they can respond in time if necessary.

1.2 Variants

The Continuous Integration process has different variants.

Version management

You have various options for versioning:

1. TIA Portal project server: Entire TIA Portal projects are saved on the project server, meaning that the hardware as well as the software objects are located there.
2. Version control system (VCS, e.g. git, Subversion, etc.): Software objects such as program blocks can be exported from TIA Portal and saved in a VCS. The export can be triggered with the Openness interface or with the TIA Portal Version Control Interface (VCI).

In this example, we will use the TIA Portal project server.

Testing

There are various options for testing:

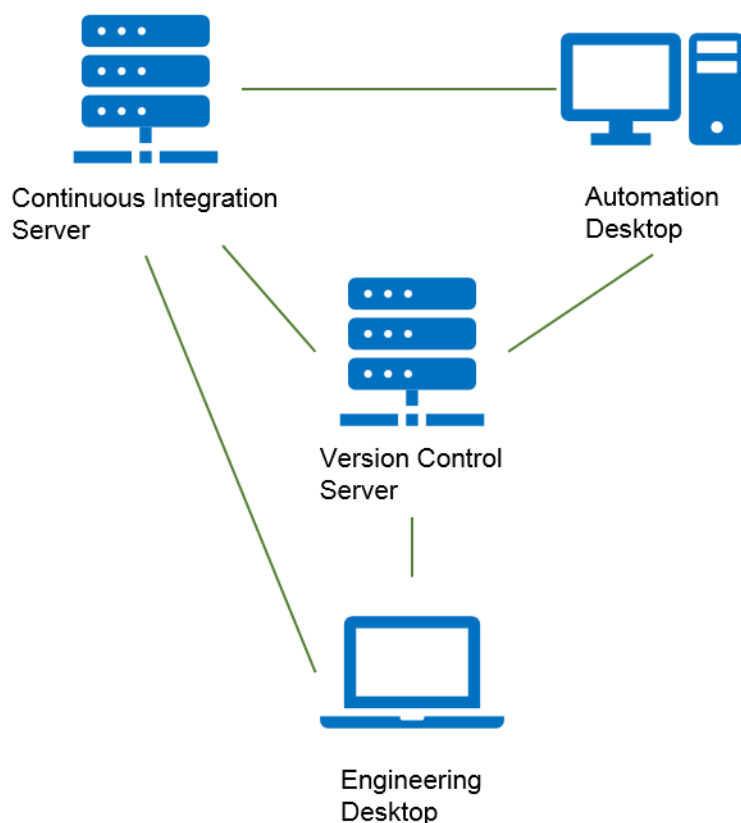
1. The application example "S7UnitTest" can create application tests based on PLCSIM Advanced.
2. The application example "Project check for TIA Portal" can be used to check the project against the programming style guide.
3. The Test Suite Advanced can perform tests against a programming style guide, in addition to application tests.

This application example is based on TIA Portal V16 and the application example "Project Check".

1.3 Principle of operation

Four different components are typically used in a Continuous Integration environment to apply the Continuous Integration workflow.

Figure 1-2: Network overview



Engineering desktop

A Microsoft Windows desktop computer, on which the user edits TIA Portal projects. Each user has his/her own engineering desktop.

Version control server

A Microsoft Windows server running the TIA Portal project server. Changes to the projects are broadcast to the Continuous Integration server and projects are exported for the automation desktop.

A version control server can be used for all projects.

Continuous integration server

A Linux or Microsoft Windows server running Jenkins. Jenkins is used to coordinate the project assembly (building) and for testing. The user can access Jenkins from his/her engineering desktop via a website.

A Continuous Integration server is needed for all projects.

Automation desktop

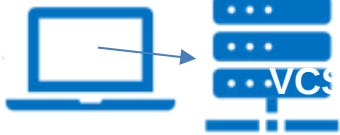
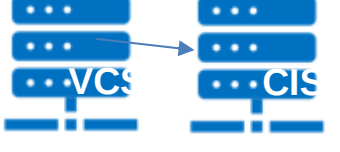
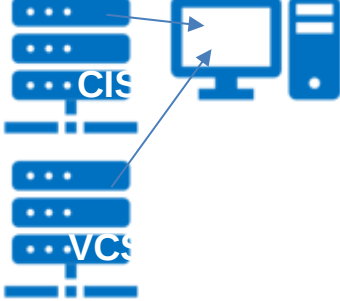
A Microsoft Windows desktop computer running a Jenkins agent. Jenkins uses this node to export and test the TIA Portal project.

You can set up multiple automation desktops to scale your test environment.

1.4 Sequence

The following sketches the sequence you will follow when you implement this application example.

Table 1-1: Sequence

	1. The user modifies a block in the TIA Portal project on his/her engineering desktop. After the change, the block is stored on the version control server as a new revision in the TIA Portal project server.
	2. In addition to the TIA Portal project server, the version control server also contains the application Project Server Listen Console, which the TIA Portal project server regularly checks for new projects and new revisions. When the application detects changes, a corresponding test job is created in the Jenkins software on the Continuous Integration server. This is accomplished with the Jenkins API.
	3. The Jenkins software on the Continuous Integration server triggers the test run on the automation desktop when batch commands are run by a connected Jenkins client on the automation desktop. With these commands it is possible to extract the project being tested from the version control server and check it using test applications, such as Project Check. The results of the tests are sent to the Continuous Integration server.
	4. In the final step, the Continuous Integration server informs the users of the engineering desktop about the result of the test.

1.5 Components used

Table 1-2: Components

Component	File name	Note
Setup documentation	109801203_Cont_Int_Setup_DOC_V13_en.pdf	CI system setup
Project Server Tools documentation	109801203_Cont_Int_ProjektServerTools_DOC_V12_en.pdf	Project Server Tools
Test Runner documentation	109801203_Cont_Int_TestRunner_DOC_V11_en.pdf	Test Runner
Transforming XML documentation	109801203_Cont_Int_Transforming_XML_DOC_V10_en.pdf	Transforming XML
Jenkins documentation	109801203_Cont_Int_Jenkins_DOC_V13_en.pdf	Jenkins
BAT files for setup	109801203_Cont_Int_Setup_PROJ_V11.zip	BAT files
Project Server Tools	109801203_Cont_Int_ProjServerTools_PROJ_V11.zip	Code, DLLs, applications
Test Runner	109801203_Cont_Int_TestRunner_PROJ_V140.zip	DLLs, applications
Transforming XML	109801203_Cont_Int_Transforming_XML_PROJ_V10.zip	Code, DLLs, applications
Readme OSS	109801203_Cont_Int_ToolboxforTIAPortal__1.0.0__READMEOSS.html 109801203_Cont_Int_TestRunner__1.0.0__ReadMe_OSS.html	Third-party software information

2 Installations

The application example "Continuous Integration with TIA Portal" consists of multiple components, each of which has its own documentation.

Overview

The following chapters describe the software that needs to be installed for the individual components in the Continuous Integration workflow.

The folder paths and access credentials in the document are only examples. You can change the folder paths according to your installation; and you must change the access credentials.

2.1 Installing the Continuous Integration server

The Continuous Integration server uses the Jenkins software system. It is used to coordinate building and testing.

2.1.1 Software

The following software is necessary on the Continuous Integration server:

Table 2-1: Jenkins components

Names	Required	Type
Continuous_Integration_mit_TIA_Portal_Jenkins.pdf	Yes	Documentation
Jenkins (LTS)	Yes	Workplace automation
Java 1.8 SDK	Yes	Framework for Jenkins
Nginx	Yes	Reverse proxy for Jenkins (for Linux installations)

2.1.2 Setting up Jenkins

Install Jenkins as described in the "Jenkins" chapter of the documentation.

2.2 Version control server installation

The version control server is used to store the projects in a centralized and version-controlled manner.

2.2.1 Software

We will use a Microsoft Windows distribution for the version control desktop.

Table 2-2: Software

Names	Required	Purpose
Project server	Yes	Source control
Project Server Listen Console from the "ProjectServerTools" application	Yes	Queries from the project server regarding new projects/revisions

2.2.2 Setting up the project server

Install the project server as described in the documentation.

Note

Documentation for setting up the project server

<https://support.industry.siemens.com/cs/ww/en/view/109740141>

2.2.3 Project Server Listen Console setup

Extract the Project Server Listen Console application from the Project Server Tools on the version control server, for example in the folder "C:\tools\ProjectServerListenConsole". Configure the Project Server Listen Console application as described in the application's documentation.

Below are the two steps for a minimal example configuration whose variables you must modify to match the environment. Configuration is accomplished with the Windows Command Prompt; enter the commands with the variables modified.

Example configuration for Jenkins data:

```
: Basic configuration incl. Jenkins files
Siemens.ProjectServerListenConsole.exe configure --
config="C:\tools\ProjectServerListenConsole\config.xml" --
jenkins=https://MyJenkinsServer/ --username="JenkinsUserID" --
password="JenkinsApiToken" --
dataPath="C:\tools\ProjectServerListenConsole\data.xml" --
templateJob="template-job-name" --templateLabel="TIAP_"
```

Example configuration for project server:

```
: Add project server configuration
Siemens.ProjectServerListenConsole.exe add --
server=http://myprojectserver.net:12345 --
config="C:\tools\ProjectServerListenConsole\config.xml"
```

The Project Server Listen Console application must regularly check the project server for new projects and revisions. To accomplish this, create the necessary **Batch file** and make it start automatically with the **Windows Autostart** function. The Batch file will run the Project Server Listen Console every 60 seconds to check and writes the responses from the application to a log file. Note that the log file can become very large. Delete it as needed.

In case of changes on the project server, the changes will be stored in the data.xml file.

Example code:

```
@echo off
:REPEAT
cls
echo The log is stored at C:\tools\ProjectServerListenConsole\listener.log
"C:\tools\ProjectServerListenConsole\Siemens.ProjectServerListenConsole.exe"
listen --config="C:\tools\ProjectServerListenConsole\config.xml" >>
"C:\tools\ProjectServerListenConsole\listener.log" 2>&1
timeout /t 60
goto REPEAT
```

This example code is supplied along with the application example as the Batch file **listener.bat**.

Note

The "Project Server Tool documentation" describes the exact configurations.

2.3 Automation desktop installation

The automation desktop is used to automatically perform work steps such the application test.

2.3.1 Software

We will use a Microsoft Windows distribution for the automation desktop. It is connected as a node with the Continuous Integration server via the Jenkins agent.

Table 2-3: Software

Names	Required	Purpose
S7-PLCSIM Advanced V3.0	Yes	Testing of TIA Portal projects
Test.Console from the application example "Project Check for TIA Portal"	Yes	Testing of projects
ProjectServerExportConsole from the application "ProjectServerTools"	Yes	Exporting projects from the project server
Transform-XML	Yes	Transform result file
Java 1.8 SDK	Yes	Framework for Jenkins agent
Jenkins agent	Yes	Connecting the node with the Jenkins server*

* not part of the application example, as it is automatically generated by Jenkins based on your configuration

2.3.2 SIMATIC PLCSIM Advanced setup

SIMATIC PLCSIM Advanced V3.0 is used to simulate the SIMATIC S7-1500 CPU in order to provide the corresponding test environment.

Install SIMATIC PLCSIM Advanced V3.0 as described in the documentation.

Note

Manual for SIMATIC S7-1500 S7-PLCSIM Advanced V3.0

<https://support.industry.siemens.com/cs/ww/en/view/109773484>

2.3.3 Setting up Project Check for TIA Portal

Project Check for TIA Portal is used to check the TIA Portal projects against a programming style guide.

Install the software as described:

Note

Project Check article and download for TIA Portal

<https://support.industry.siemens.com/cs/ww/en/view/109741418>

2.3.4 Setting up Project Server Export Console

The Project Server Export Console application will be needed to export the projects from the project server for testing.

Extract the Project Server Export Console application from the Project Server Tools on the test machines (automation desktop), for example in the folder "C:\tools\ProjectServerExportConsole".

2.3.5 Setting up Transforming XML

Transforming XML is used to transform the results from various applications (e.g. Project Check for TIA Portal) into the JUNIT format that Jenkins reads natively.

Extract the Transforming XML application on the test machines (automation desktop), for example in the folder "C:\tools\Transforming-XML".

2.3.6 Execution environment for Jenkins

Certain preparations have to be made so that Jenkins can automatically start the tests on the automation desktop.

To start automated processes in the most flexible manner possible, we recommend the use of one or more Batch files. Here is how you can keep the function calls in Jenkins compact and offload the logic to the test machines:

- **ProjectExport.bat** for exporting projects from the project server to make them ready for the test environment
- **StyleCheck.bat** for checking projects, for example with the application Project Check or Test Suite in TIA Portal

Other Batch files, for example for application testing or similar, are also possible.

The following will list two examples of Batch files where you may have to modify the paths (marked in blue) to match your installation.

Example code:

```
set serverUrl=%~1
set projectName=%~2
set revisionId=%~3
set WORKSPACE=%~4
set ERRORLEVEL=%~5
set COMPUTERNAME=%~6

@echo off
mkdir "%WORKSPACE%\logs"
echo Building...
"C:\tools\ProjectServerExportConsole\Siemens.ProjectServerExportConsole.exe"
export --server="%serverUrl%" --project="%projectName%" --
revisionId=%revisionId% --targetDir="%WORKSPACE%"
echo %ERRORLEVEL%
rename "%WORKSPACE%\%projectName%" project
rename "%WORKSPACE%\project\*.ap16" project.ap16
```

This example code is supplied along with the application example as the Batch file **ProjectExport.bat**. When executed, the requested project will be exported via the Project Server Export Console application; and the name and directory of the project will be harmonized.

Depending on the handling with the Project Check application or Test Runner, one of the following two Batch files will be used.

Example code for using the "Project Check" application

```
set serverUrl=%~1
set projectName=%~2
set revisionId=%~3
set WORKSPACE=%~4
set ERRORLEVEL=%~5
set COMPUTERTNAME=%~6

@echo off
echo Testing...

"C:\tools\ProjectCheck\Siemens.ProjectCheckConsole.exe" -p
"%WORKSPACE%\project\project.ap16" -l "%WORKSPACE%\logs\projectcheck.log" -r
"%WORKSPACE%\logs\projectcheck.report.xml"
"%WORKSPACE%\logs\projectcheck.report.xlsx"

"C:\tools\transforming-xml\bin\xslt.exe" -i
"%WORKSPACE%\logs\projectcheck.report.xml" -o
"%WORKSPACE%\logs\projectcheck.junit.xml" -x "C:\tools\transforming-
xml\stylesheets\projectcheck2junit.xsl" -p Hostname:"%COMPUTERTNAME%"

echo %ERRORLEVEL%
```

This example code is supplied in the application example as the Batch file **StyleCheck_ProjectCheck.bat**. When executed, the Project Check application will be run on the exported project, and the result will be converted to the JUNIT format with the Transforming XML application.

Note

If you intend to use the "Project Check" application, the file "**StyleCheck_ProjectCheck.bat**" must be renamed to "**StyleCheck.bat**".

Example code for using Test Suite for style guide and application tests

```
set serverUrl=%~1
set projectName=%~2
set revisionId=%~3
set WORKSPACE=%~4
set ERRORLEVEL=%~5
set COMPUTERTNAME=%~6

@echo off
echo Testing...

"C:\tools\TestRunner\Siemens.TestRunner.Console.exe" --debug --logfile
%WORKSPACE%\logs\testrunner.log --report %WORKSPACE%\logs\testrunner.junit.xml
--format=JUNIT_HUDSON --project=%WORKSPACE%\project\project.ap17 --test-
application=all --test-style=all

echo %ERRORLEVEL%
```

This example code is supplied along with the application example as the Batch file **StyleCheck_TestSuite.bat**. When executed, the test cases in Test Suite will be run on the exported project, and the results will be reported. Calling of Test Suite can be automated with TIA Portal V17 or later.

Note

If you intend to use the Test Suite test cases, the file "**StyleCheck_TestSuite.bat**" must be renamed to "**StyleCheck.bat**".

The Batch file can also be split up so that one Batch file calls the style guide tests (e.g. runs at each new revision) and another Batch file calls the application tests (e.g. runs nightly in case of new revisions). If this is the case, bear in mind that the log files must have different names if the split-up Batch files are run in one build job. This prevents log files from being overwritten.

Note

The "Test Runner documentation" describes the exact configurations.

2.3.7 Jenkins agent

So that Jenkins can run the commands on the respective test machines (automation desktop), the test machines must be connected with Jenkins. This is accomplished with the so-called Jenkins agents, which you can launch via the website for your Jenkins installation when selecting the corresponding node.

For more information on installation, refer to the chapter entitled "Connecting Jenkins agents" in the documentation:

Java environment variable

To run the Jenkins agent, a Java environment variable must be created by the installation program. If the Jenkins agent is not started, check whether the environment variable was set; create it if necessary. Navigate to the Control Panel, then to System Properties, then Environment Variables. If necessary, enter the environment variable "JAVA_HOME" with the path from the Java installation directory.

Figure 2-1: System settings

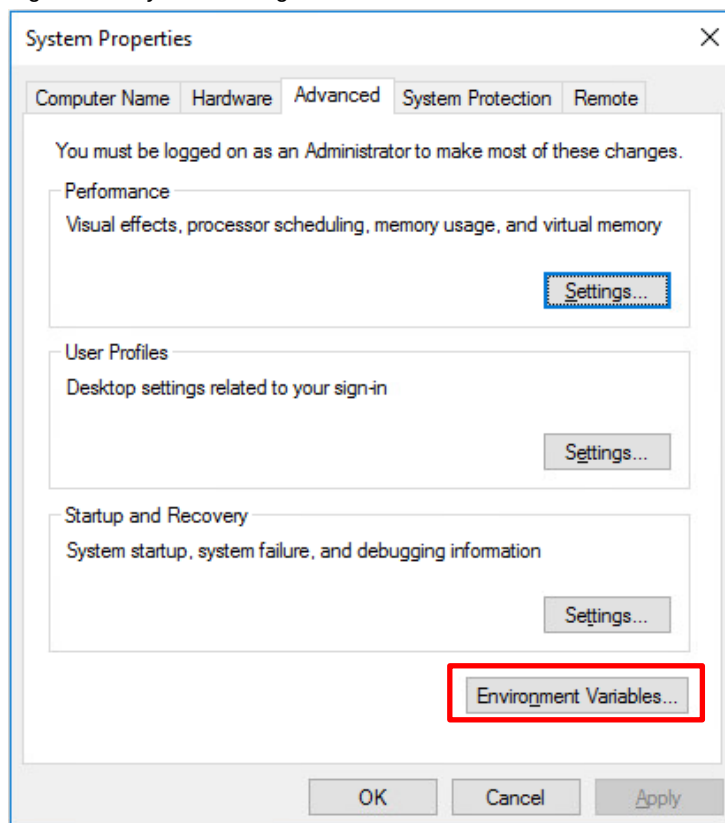
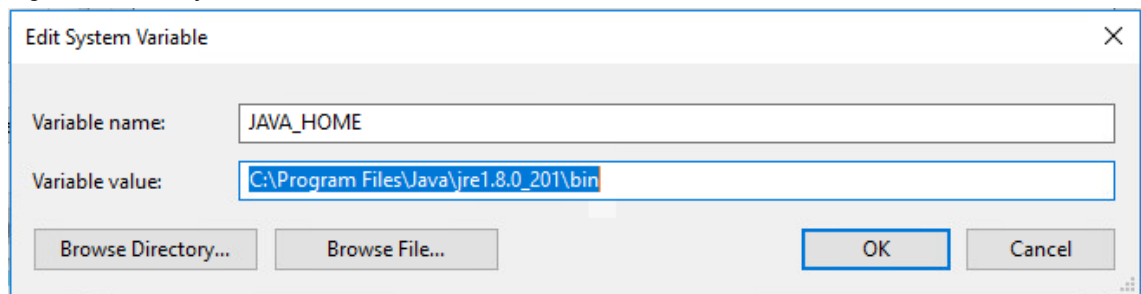


Figure 2-2: Edit System Variable



2.4 Engineering desktop installation

2.4.1 Software

We will use a Microsoft Windows distribution for the engineering desktop.

Table 2-4: Software

Names	Required	Purpose
.NET Framework 3.5	Yes	Framework for TIA Portal
TIA Portal	Yes	Creating TIA Portal projects

2.4.2 TIA Portal setup

TIA Portal is used to create and modify automation projects.

Install TIA Portal as described in the documentation.

Note

Manual for SIMATIC STEP 7 Basic/Professional V16

<https://support.industry.siemens.com/cs/ww/en/view/109773506>

Manual for SIMATIC STEP 7 Basic/Professional V17

<https://support.industry.siemens.com/cs/ww/en/view/109798671>

3 Appendix

3.1 Service and support

Industry Online Support

Do you have any questions or need assistance?

Siemens Industry Online Support offers round the clock access to our entire service and support know-how and portfolio.

The Industry Online Support is the central address for information about our products, solutions and services.

Product information, manuals, downloads, FAQs, application examples and videos – all information is accessible with just a few mouse clicks:

support.industry.siemens.com

Technical Support

The Technical Support of Siemens Industry provides you fast and competent support regarding all technical queries with numerous tailor-made offers

– ranging from basic support to individual support contracts. Please send queries to Technical Support via Web form:

siemens.com/SupportRequest

SITRAIN – Digital Industry Academy

We support you with our globally available training courses for industry with practical experience, innovative learning methods and a concept that's tailored to the customer's specific needs.

For more information on our offered trainings and courses, as well as their locations and dates, refer to our web page:

siemens.com/sitrain

Service offer

Our range of services includes the following:

- Plant data services
- Spare parts services
- Repair services
- On-site and maintenance services
- Retrofitting and modernization services
- Service programs and contracts

You can find detailed information on our range of services in the service catalog web page:

support.industry.siemens.com/cs/sc

Industry Online Support app

You will receive optimum support wherever you are with the "Siemens Industry Online Support" app. The app is available for iOS and Android:

support.industry.siemens.com/cs/ww/en/sc/2067

3.2 Industry Mall



The Siemens Industry Mall is the platform on which the entire Siemens Industry product portfolio is accessible. From the selection of products to the order and the delivery tracking, the Industry Mall enables the complete purchasing processing – directly and independently of time and location:

mall.industry.siemens.com

3.3 Links and literature

Table 3-1

No.	Topic
\1\	Siemens Industry Online Support https://support.industry.siemens.com
\2\	Link to this entry page of this application example https://support.industry.siemens.com/cs/ww/en/view/109801203
\3\	TIA Portal documents https://support.industry.siemens.com/cs/ww/en/view/65601780
\4\	TIA Portal Test Suite - manual https://support.industry.siemens.com/cs/ww/en/view/109779356
\5\	Project Check for TIA Portal: Check against programming style guides https://support.industry.siemens.com/cs/ww/en/view/109741418

3.4 Change documentation

Table 3-2

Version	Date	Modifications
V1.0	08/2021	First version
V1.1	11/2021	Modifications in the setup batch files and documentation
V1.2	05/2022	Modifications in the setup batch files and documentation
V1.3	03/2023	Added to Batch file documentation