Installing WebSpellChecker on Windows with Apache Tomcat

This step-by-step guide is intended to outline the main steps for the WebSpellChecker application installation on a Windows Server with Apache Tomcat Application Server. In this guide we will run you through the whole installation process from downloading installation packages to embedding spell /grammar checking functionality to your web-based system.

⊘ **Before you begin:**

1. Check if your environment configuration meets Installation requirements.
2. Acknowledge with Architecture diagram.
3. Make sure that you have a valid license:

   First of all you need to get a valid License Ticket ID to proceed with the license activation. Depending on the server license type, it may be a commercial (e.g. 1 year) or a 30-day trial license. Except the validity period, there is no difference between ticket types from the technical perspective.

   Good to know:

   - If you do not have a license yet, you can obtain it using one of the following ways:
     - request a 30-day trial license for evaluation and testing purposes;
     - acquire a commercial license by contacting our sales department.
   - Starting from v5.8.1 released on December 30, 2020, the license is no longer validated by or tied to the hardware characteristics of a machine or server. License for older versions of the packages is hardware-dependent.
   - If you are migrating WebSpellChecker from one server to another, you will need to deactivate (detach) a license on the current server and then reactivate it on a new one. Check carefully the **Migrating License to New Server** guide under the Licensing section.
   - If you have problems with your license, contact us.

1. Download Installation Package

Download WebSpellChecker application package for a corresponding version of your operating system (OS). By default, it is provided on a license purchase or maintenance renewal. You can also request it contacting us at support@webspellchecker.net.
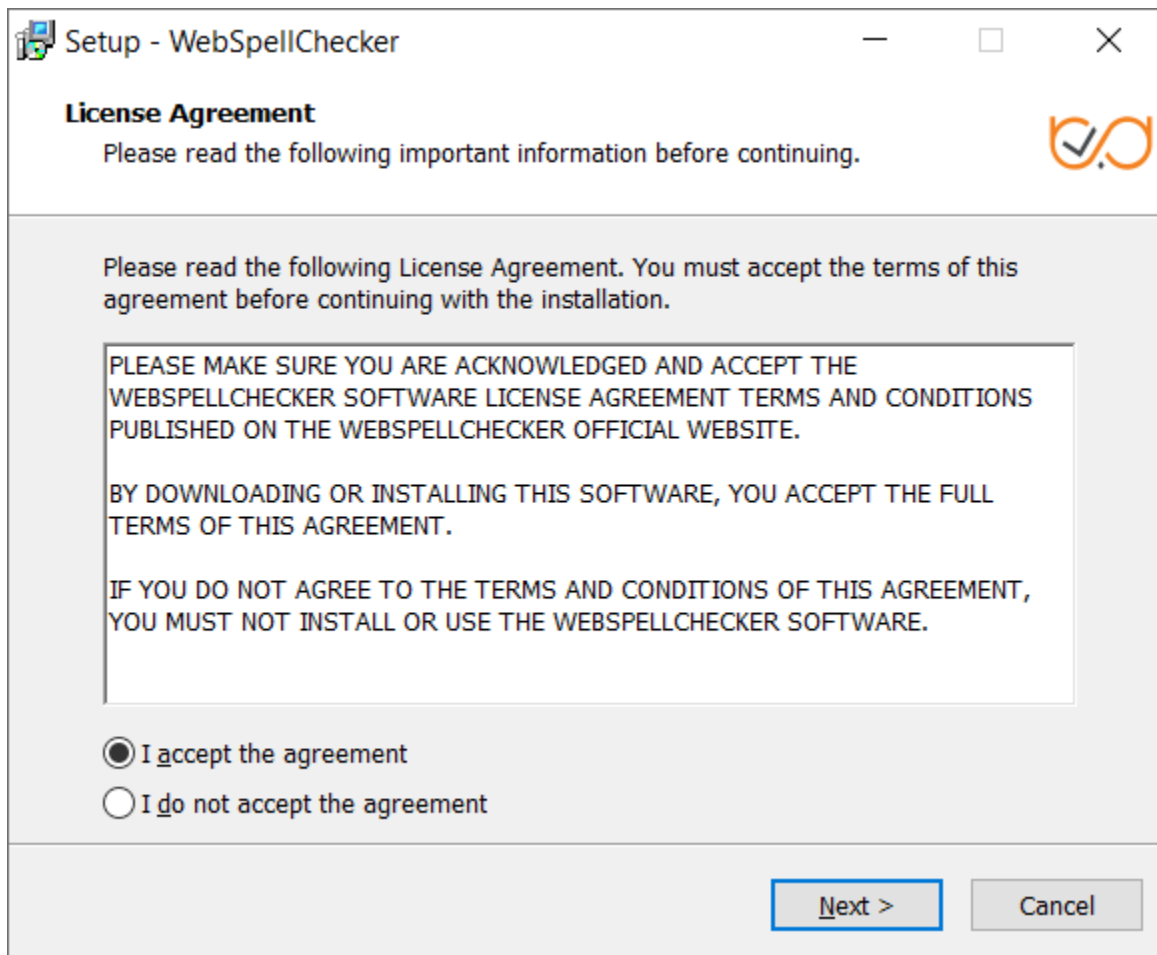
2. Install WebSpellChecker

⚠ Follow the instructions in the setup window, selecting the options that are appropriate to your environment. It is recommended that you install the AppServer as a service. To do so, select the **Install Application Server as service** checkbox.

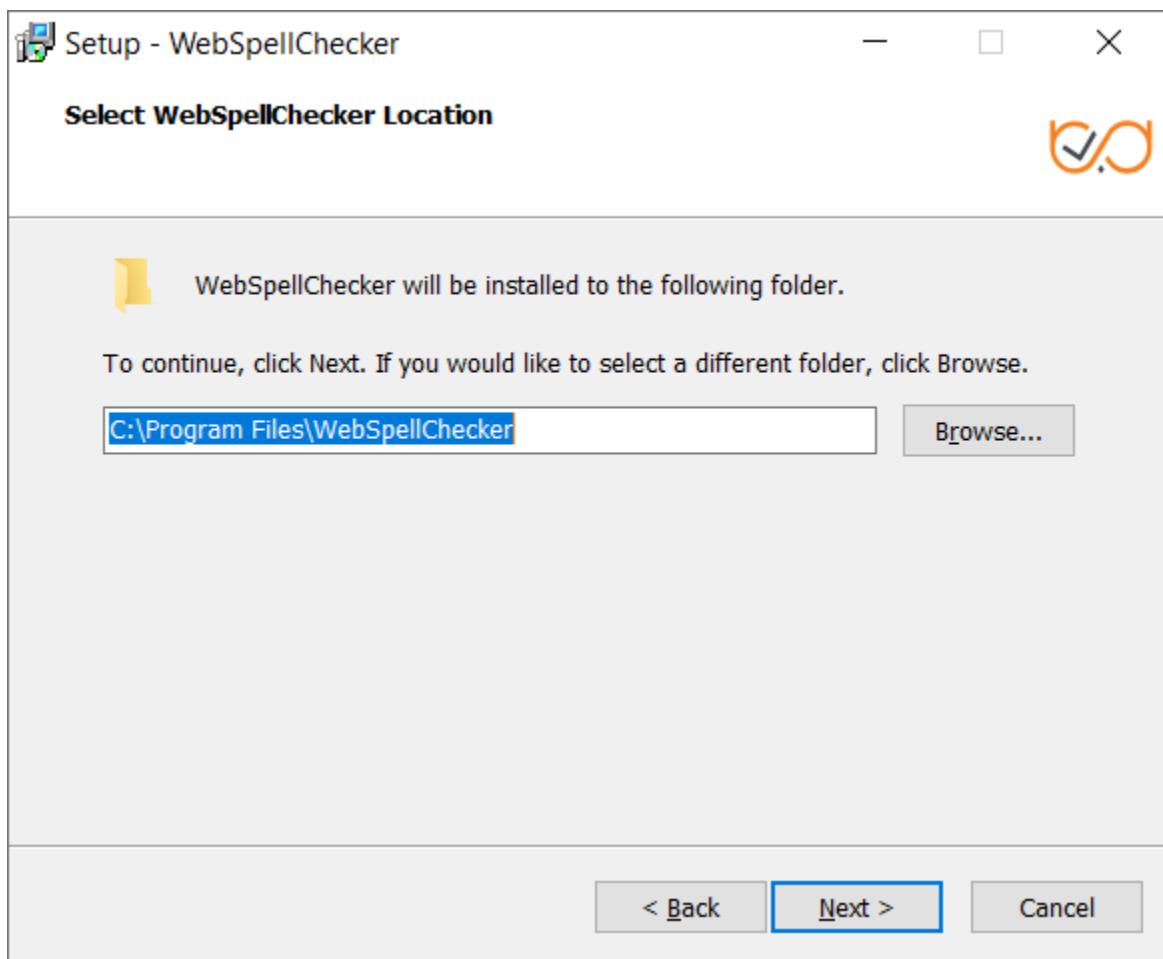2.1. Run the installation package (e.g. wsc_app_win_x64_release_5.5.x.x_xx.exe).

2.2. Before proceeding with the installation, read and confirm **WebSpellChecker Software License Agreement**.
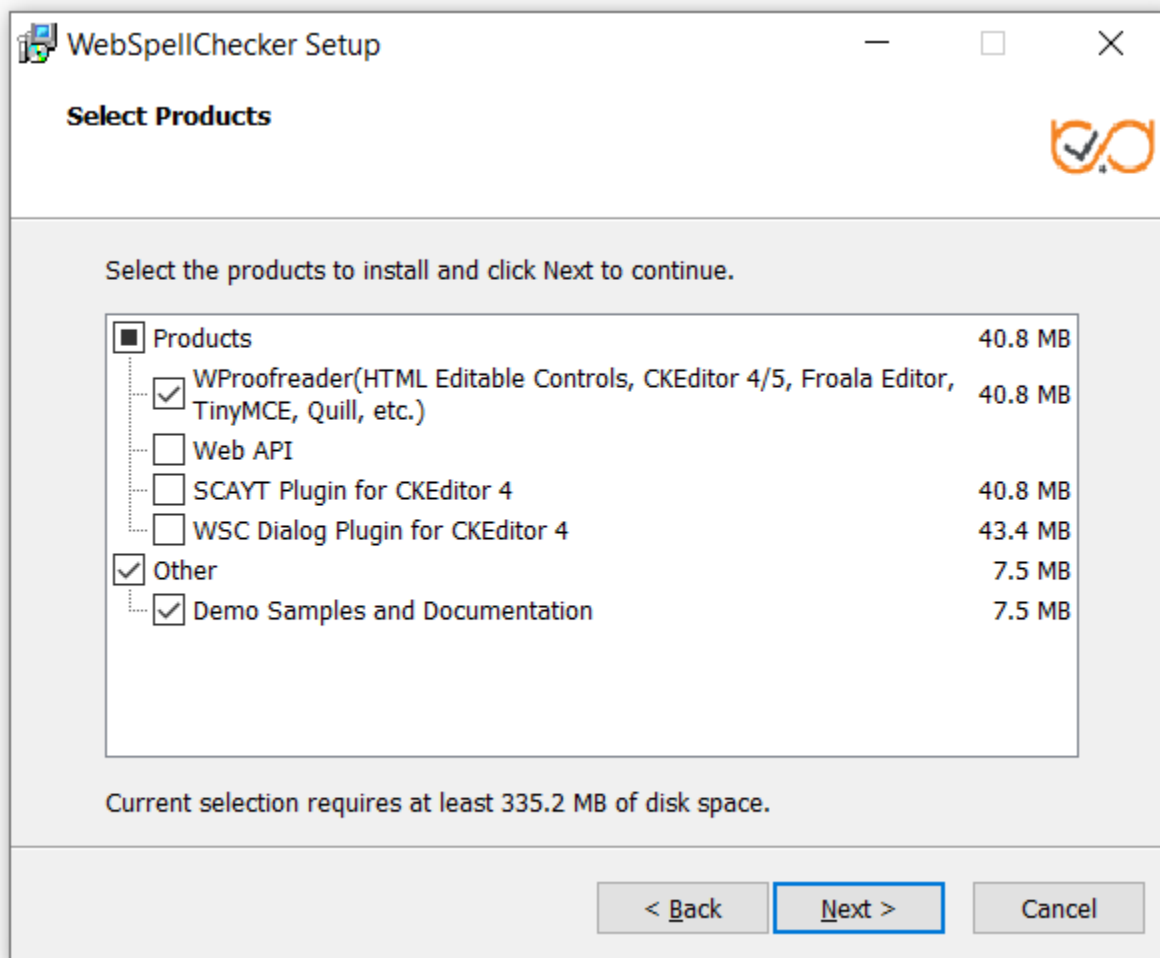
⊘ Visit WebSpellChecker Software License Agreement page to acknowledge with the full terms and conditions of WebSpellChecker Software.

Setup - WebSpellChecker

License Agreement

Please read the following important information before continuing.

Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.

> PLEASE MAKE SURE YOU ARE ACKNOWLEDGED AND ACCEPT THE WEBSPELLCHECKER SOFTWARE LICENSE AGREEMENT TERMS AND CONDITIONS PUBLISHED ON THE WEBSPELLCHECKER OFFICIAL WEBSITE.
>
> BY DOWNLOADING OR INSTALLING THIS SOFTWARE, YOU ACCEPT THE FULL TERMS OF THIS AGREEMENT.
>
> IF YOU DO NOT AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT, YOU MUST NOT INSTALL OR USE THE WEBSPELLCHECKER SOFTWARE.

◉ I accept the agreement

◯ I do not accept the agreement

Next >   Cancel

2.3. Select WebSpellChecker installation location and click **Next** to proceed. The default WebSpellChecker installation folder is **C:\Program Files\WebSpellChecker**. Click **Browse…** and specify a different installation location if it is necessary.
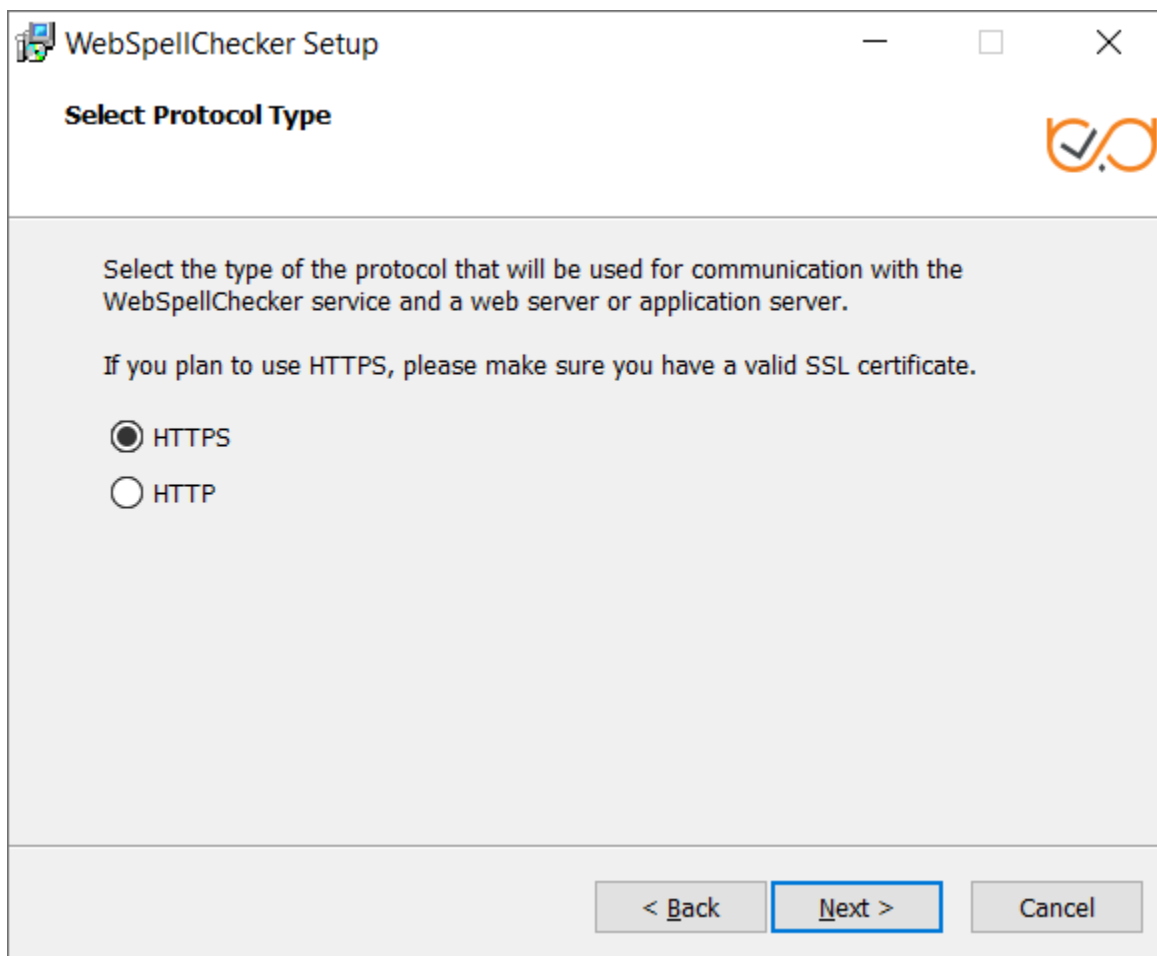
2.4. Select the products to install and click **Next** to proceed. By default, WProofreader as well as Demo Samples and Documentation are installed.

⚠ If you selected the WSC Dialog Plugin for CKEditor 4, all the requests and static files will be served by a selected web server or Java application server via the FastCGI protocol (SSRV.CGI component) which you need to enable later. For details, refer to WebSpellChecker diagram description. All other products communicate with AppServer directly and don't require any additional configuration and enabling FastCGI protocol.

2.5. Select the type of protocol that will be used for communication with the WebSpellChecker service and a web server or Java application server and click **Next** to proceed. By default, **HTTPS** is selected.

WebSpellChecker Setup — □ ✕

Select Protocol Type

Select the type of the protocol that will be used for communication with the WebSpellChecker service and a web server or application server.

If you plan to use HTTPS, please make sure you have a valid SSL certificate.

◉ HTTPS

○ HTTP

< Back   Next >   Cancel

2.6. If you have selected HTTPS on the previous step, specify the path to a valid SSL certificate file (*.pfx file) and type its password to establish a secure connection. Otherwise, this step is not available.

⚠ You can **skip** this step using the checkbox and use the SSL certificate installed on your web or Java application server. In this case the connection with the service will be configured via the FastCGI protocol.

2.7. Specify the service port. The default port of AppServer is 2880. Press **Next** to proceed with the installation using port 2880.

WebSpellChecker Setup

Specify Service Port

Specify the service port (by default: 2880). If you prefer to use a custom port, make sure it is not in use by another service.
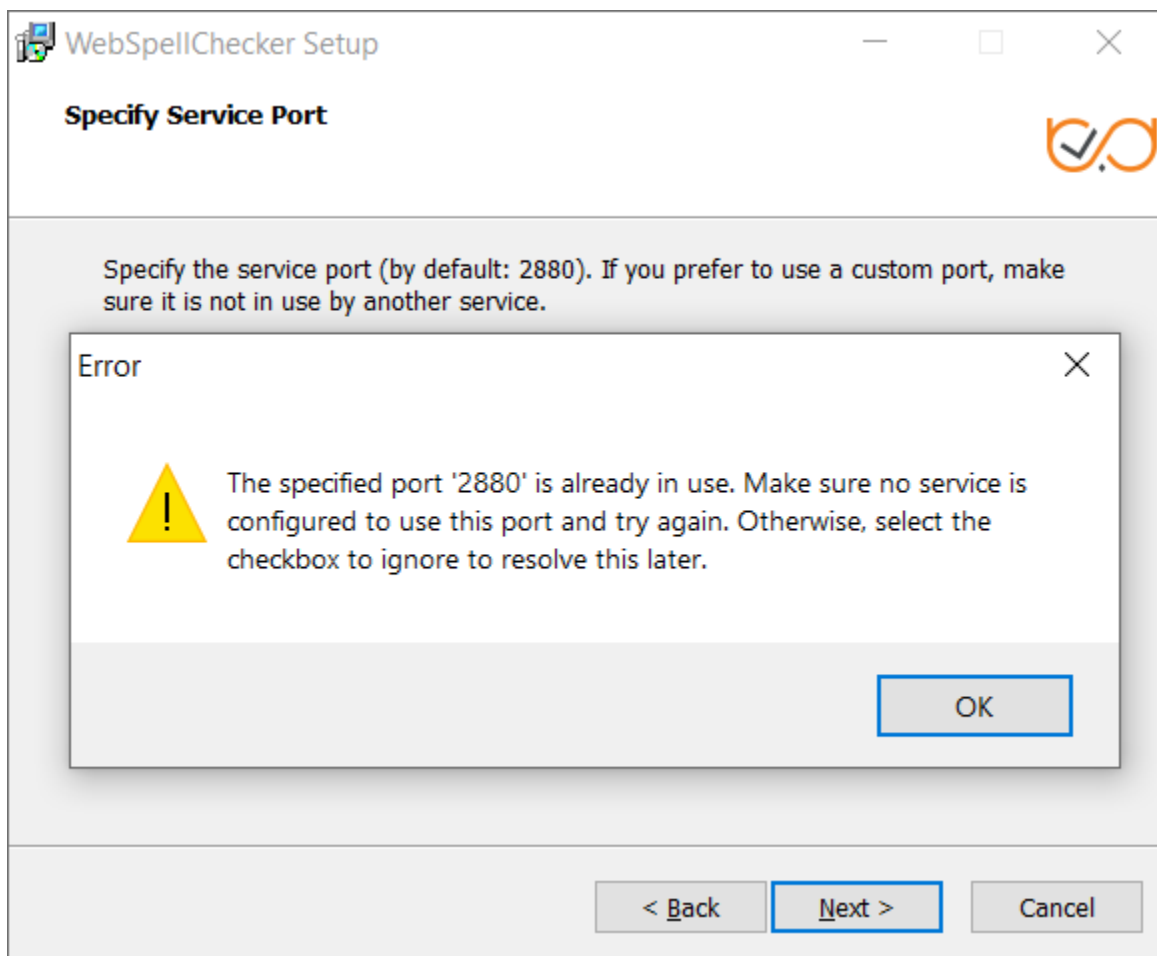
2880

☑ Ignore and use this port even if it is in use.

You have an option to use the specified port for the setup even if it is in use by another service. However, you will be required to manually resolve the issue.

< Back   Next >   Cancel

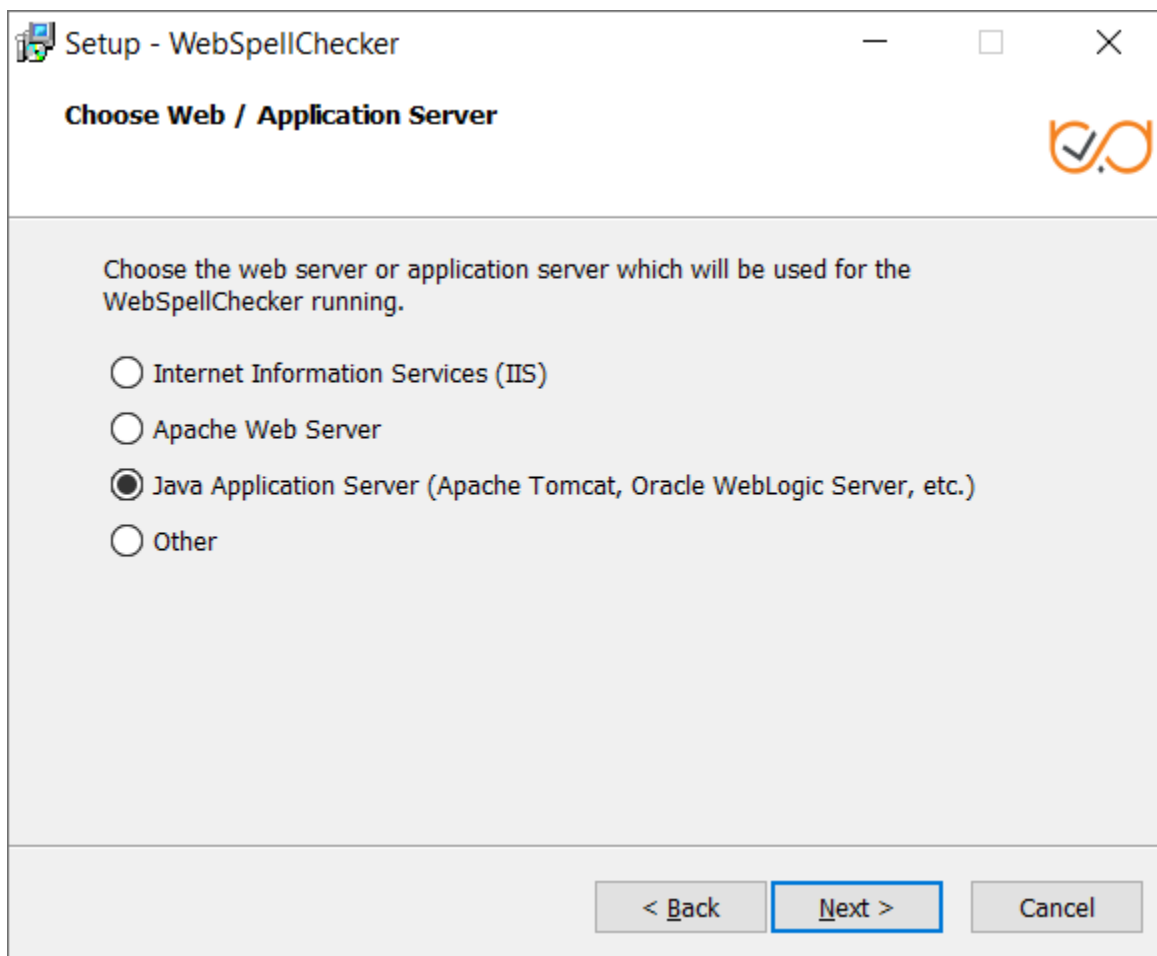If the port is already in use, the following error message appears.

To continue the installation under the port 2880, press **OK** and select the **Ignore and use this port even if it is in use** option.

> ⚠ If you also have an installation of WebSpellChecker Server on the same server, and AppServer is running under the port 2880, you need to select a different unused port (e.g. 2881) for this installation.

2.8 Select **Java Application Server (Apache Tomcat, Oracle WebLogic Server, etc.)** as a Java application server and configure its settings.

⚠ This section outlines the details of wscservice.war deployment on Apache Tomcat (static files and all requests to engines are served) or binding only static files and the rest of the service requests such as spelling or grammar checking, directly connecting with AppServer.

Option A: Binding of Static Files and Serving Service Requests by AppServer

To bind only static files and the rest of the service requests such as spelling or grammar checking directly connecting with AppServer:

1. Stop Apache Tomcat before modifying the configuration files.

2. Add the following paths to the **server.xml** file inside the **<Host name="localhost" appBase="webapps" unpackWARs="true" autoDeploy="true">** tag:
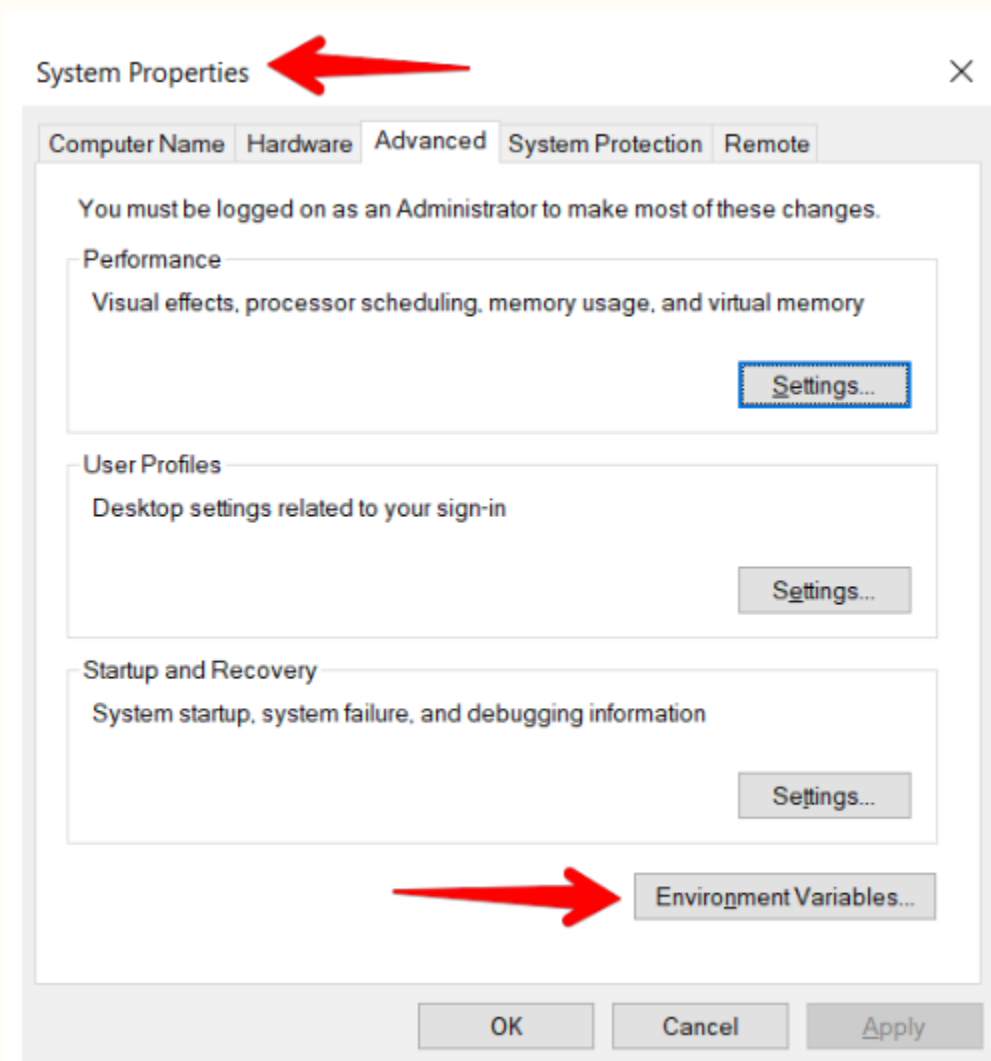
```
<Context docBase="C:/Program Files/WebSpellChecker/WebComponents/WebInterface" path="/wscservice" />
<Context docBase="C:/Program Files/WebSpellChecker/WebComponents/Samples" path="/wscservice/samples"
/>
```

3. If you plan to host WebSpellChecker and your web app(s) on different domains, you might face the CORS issues. To avoid this, add the **Access-Control-Allow-Headers** to enable cross-origin requests to the Apache Tomcat settings. For this you can use CORS filter.

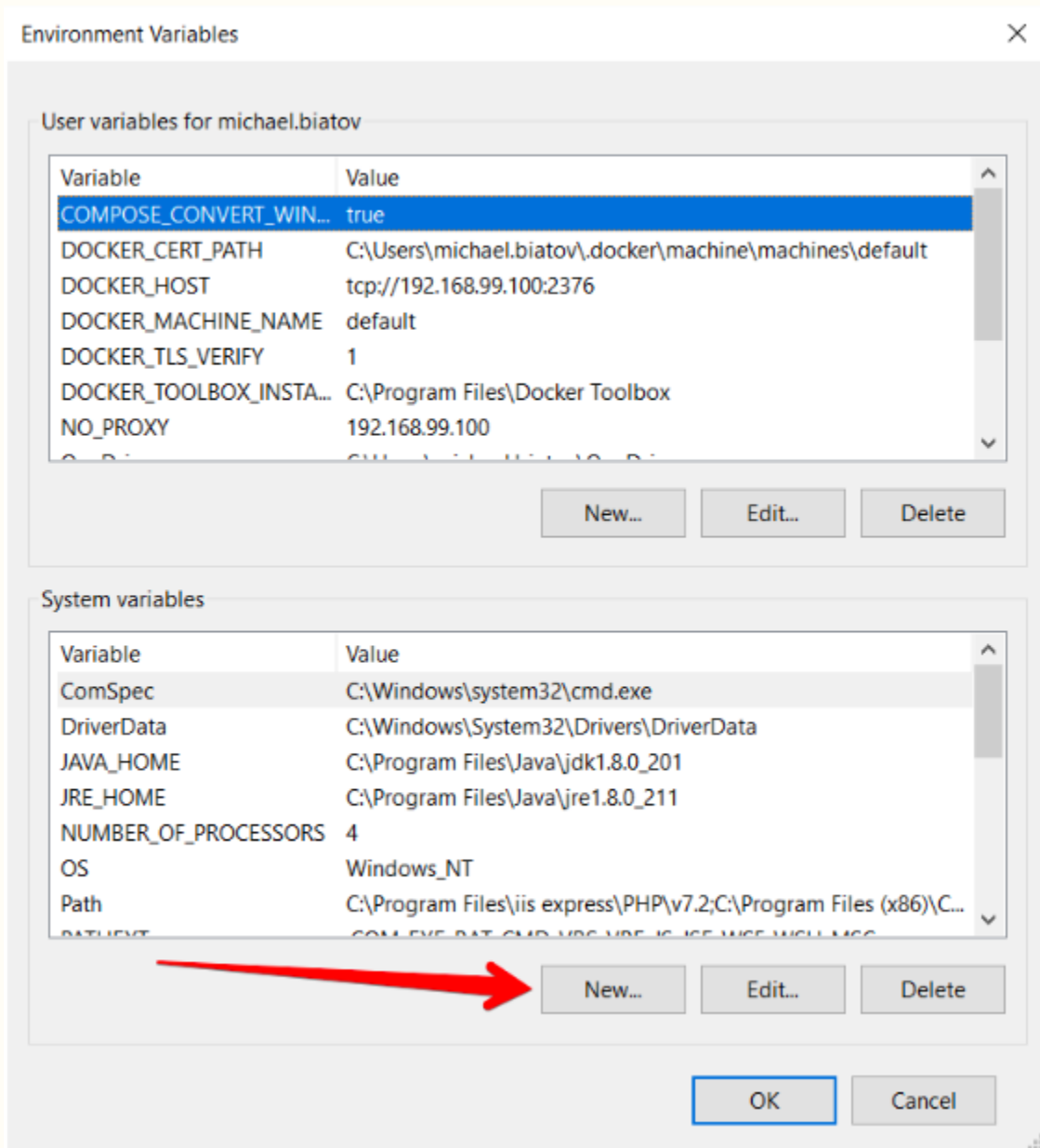4. Start Apache Tomcat after modifying the configuration files.

Option B: Serving both Static Files and Service Requests by Java Application Server (deployment of wscservice.war)

To deploy **wscservice.wa**r on Apache Tomcat for binding static files and all service requests to engines such as spelling or grammar checking:

1. Create the **WSC_DIR** environment variable. It must be pointed to the WebSpellChecker Server installation directory.

   - Go to Apache Tomcat System Properties. Open the **Advanced** tab. In this tab, locate the **Environment Variables...** button.

- Go to the section with **System variables.** In this section, click **New…** to create a new variable.

- A new pop-up window **New System Variable** will be open. Type a variable name and variable value as follows:

New System Variable

Variable name:	WSC_DIR
Variable value:	C:\Program Files\WebSpellChecker

Browse Directory...   Browse File...   OK   Cancel

2. Deploy **wscservice.war** on your Apache Tomcat Application Server.

- Copy the **wscservice.war** file from <WSC_DIR>/J2EEServerComponents to the Apache Tomcat **/webapps** directory.
- Start Apache Tomcat, or stop and then start if it is already running.

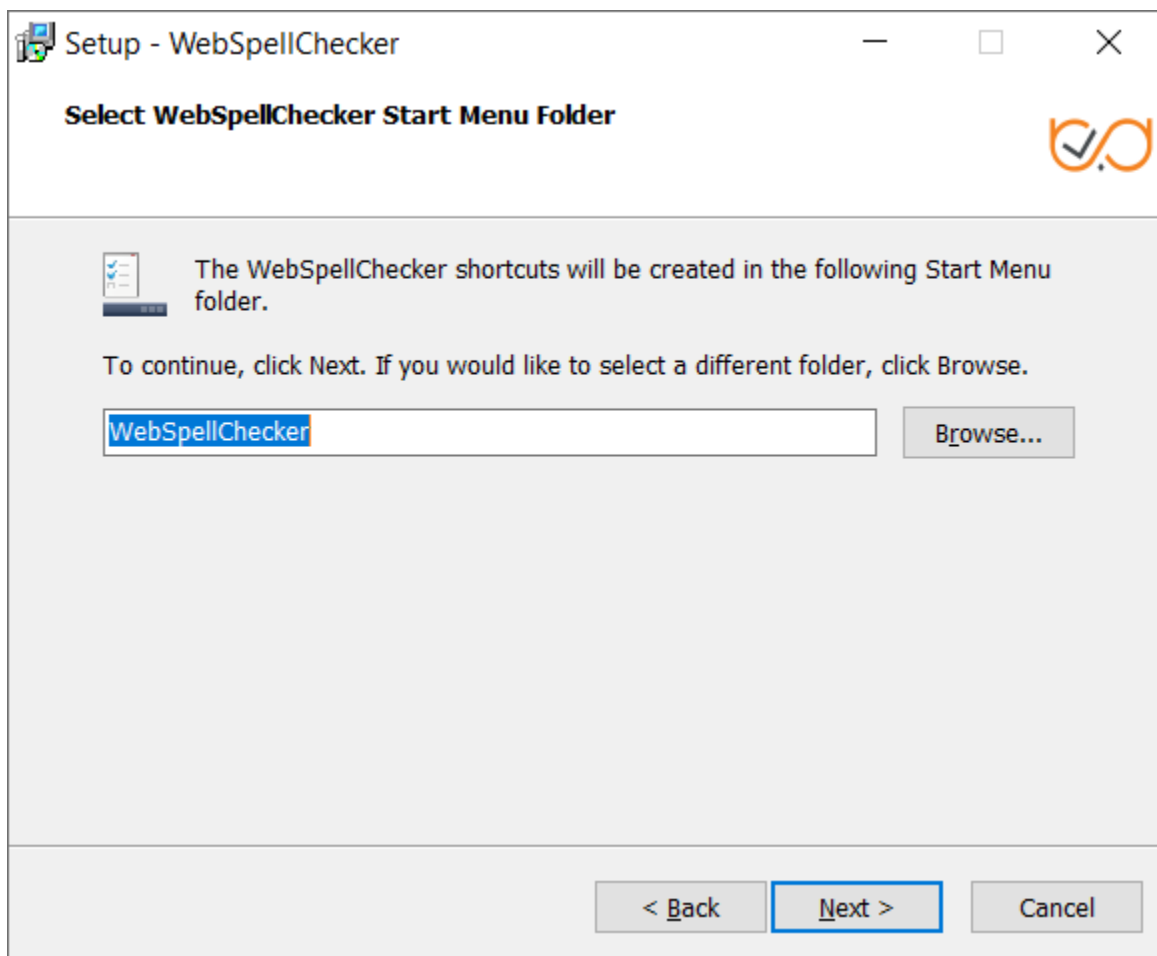✅ Note that the **wscservice.war** file should be unpacked automatically. Otherwise, unpack it manually.

2.9. Define settings of the selected Java application server and click **Next** to proceed.

- Specify a domain name (e.g. localhost).
- Specify a port number of a Java application server. The installer will suggest the port number based on the specified protocol and selected Java application server. The default ports for Java application servers are 8080 for HTTP and 8443 for HTTPS protocols.
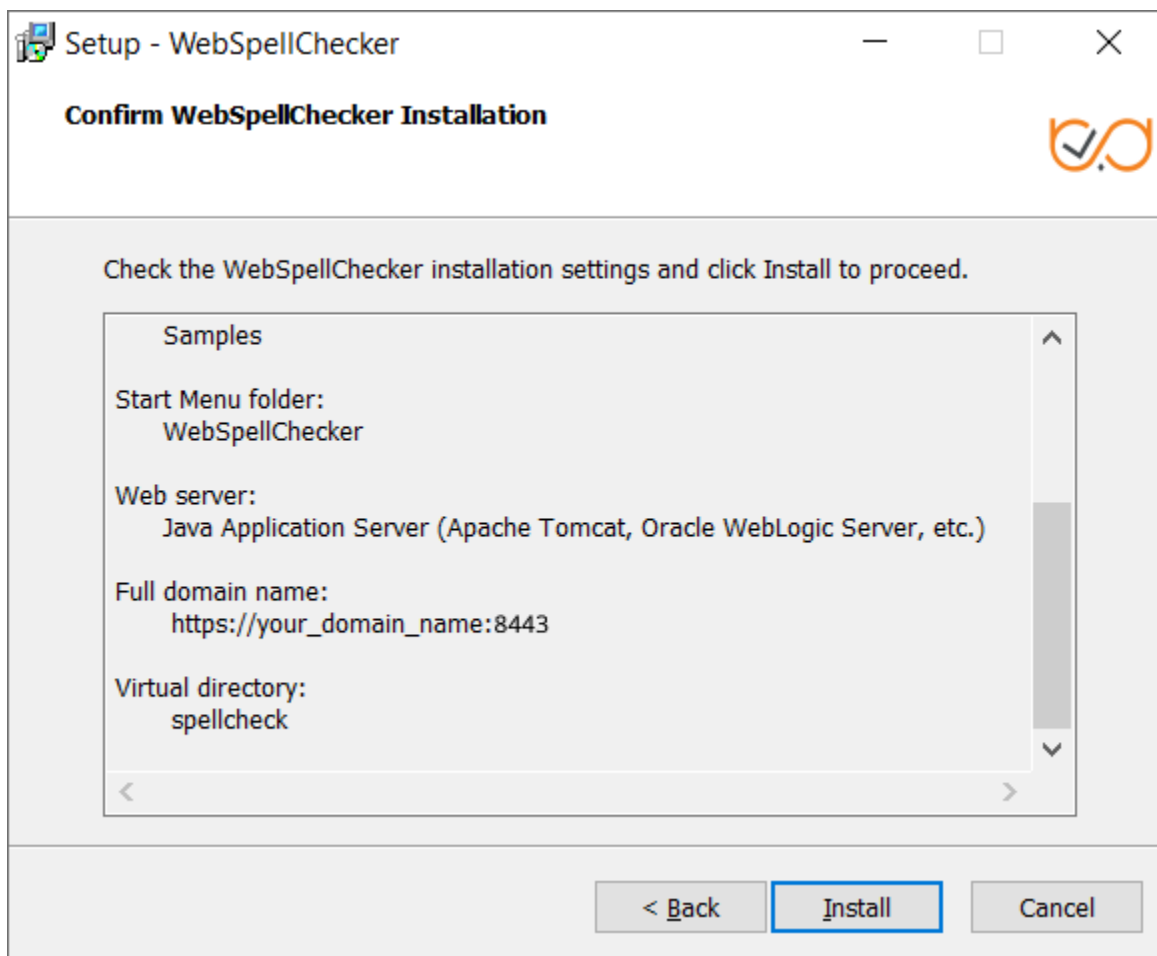
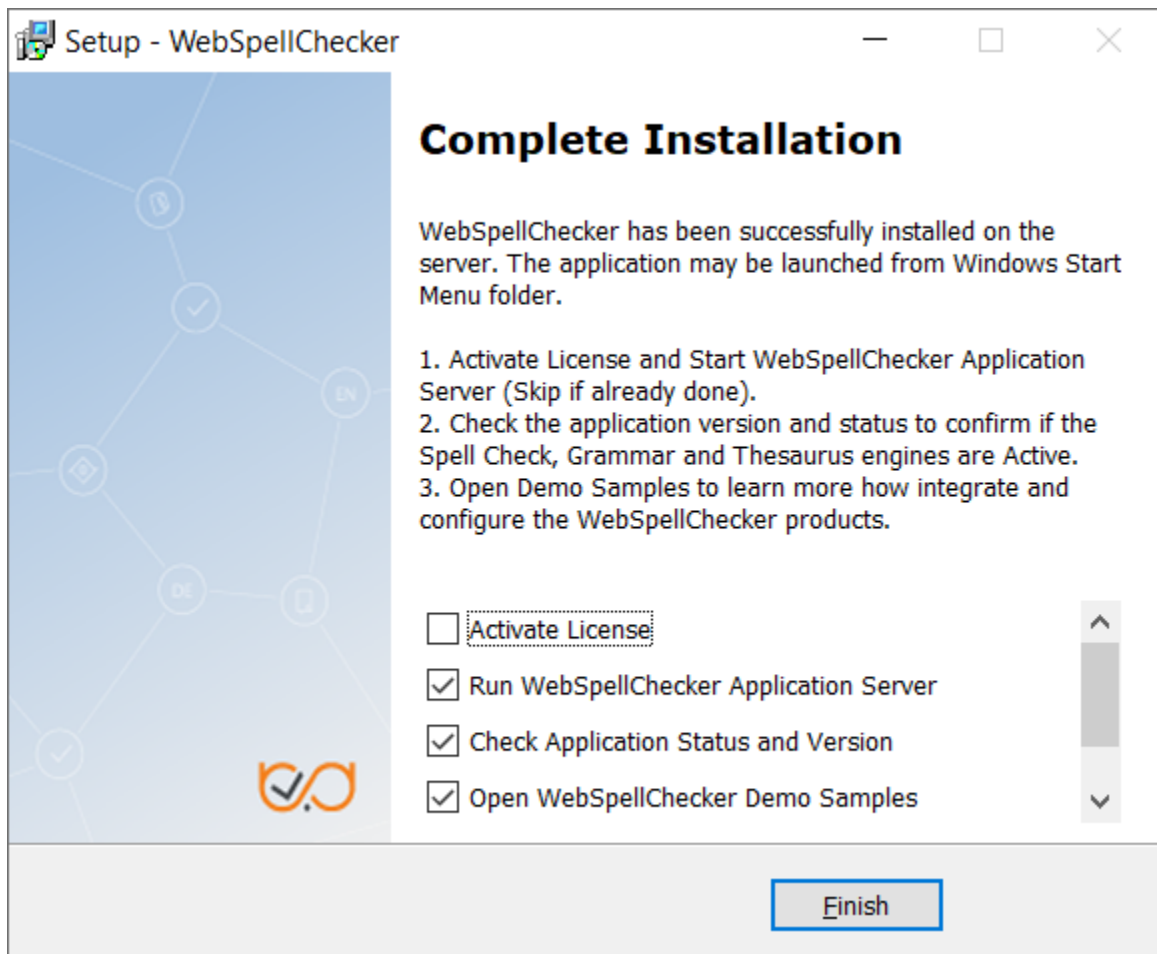Provide the name of the virtual directory for WebSpellChecker service. The default value is **wscservice**.

2.10. Select WebSpellChecker **Start Menu folder** and type a program name which will be used as a Start Menu item and click **Next** to proceed.

Setup - WebSpellChecker

Select WebSpellChecker Start Menu Folder

The WebSpellChecker shortcuts will be created in the following Start Menu folder.

To continue, click Next. If you would like to select a different folder, click Browse.

WebSpellChecker   Browse...

< Back   Next >   Cancel

2.11. On this step the installer summarizes the information that you have entered on the previous steps. Click **Install** to proceed with the installation process or return to previous steps if you want to make any changes.

2.12. As soon as the installation process is finished, select **Activate License** if you are installing WebSpellChecker Server for the first time and then click **Finish** to close the installation dialog. You can view the samples and verify application operability in newly opened web browser tabs, namely Samples page, a page with Status check, and a page with Version check. Note that all these options are selected by default. The **Activate License** option is not selected only if you reinstall WebSpellChecker on the same server.
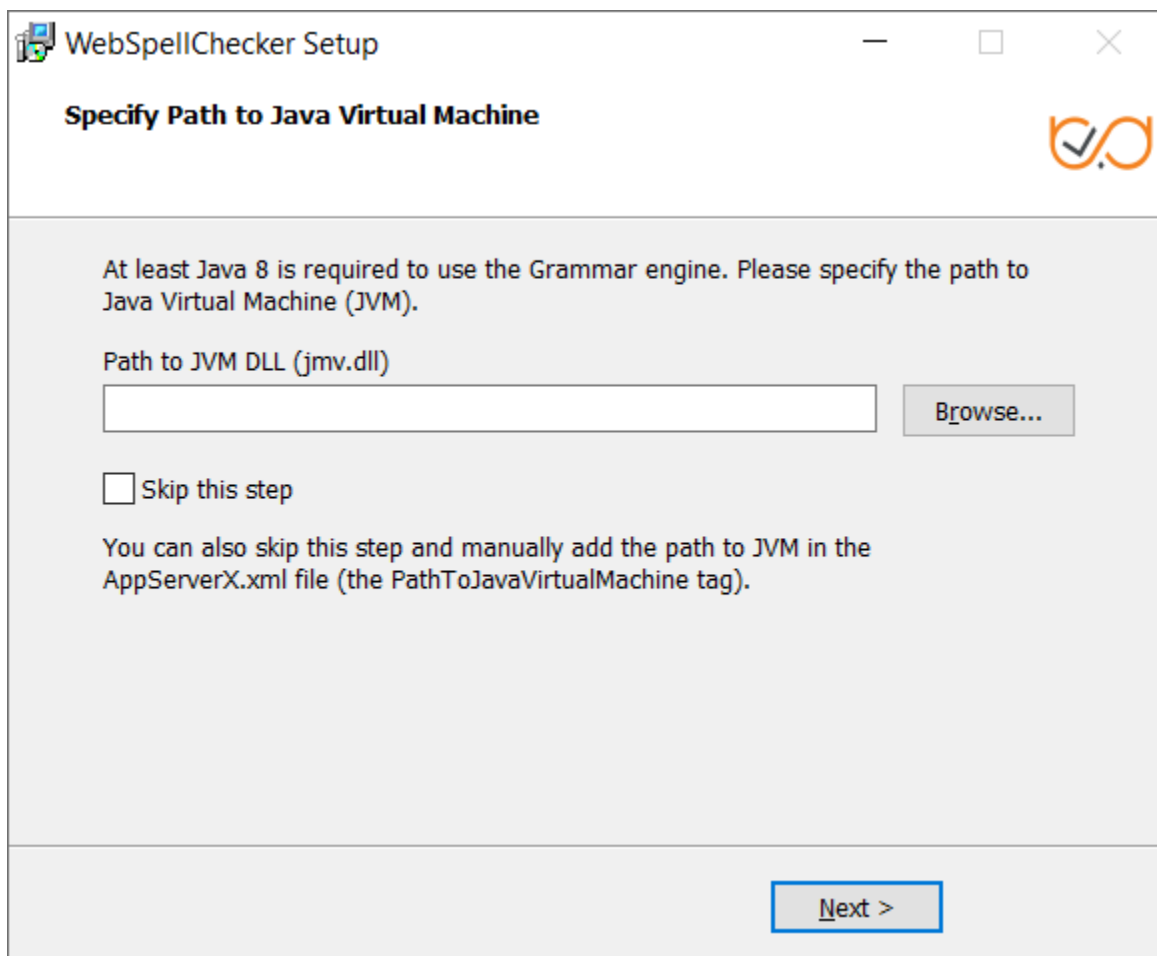
3. Activate License

Refer to the License Activation on Windows guide if you are installing WebSpellChecker Server for the first time to find out how to activate a license on a Windows-based server.

4. Configure Grammar Engine

The Grammar engine requires **64-bit version** of Java 8 or higher. The installation wizard will try to auto-detect a path to an existing Java Virtual Machine (JVM). If no path is found, you will be asked to specify the path to JVM using the next window:

You have an option to set a path to JVM manually or change the auto-detected path to your custom one. Refer to the Enabling Grammar Engine to find out more.

5. Verify Operability

Check the status and version of AppServer to verify if WebSpellChecker Server works properly. To do so, use the links below to navigate to the detailed commands and responses descriptions:

- Check AppServer version;
- Check AppServer status.

6. Integrate WebSpellChecker

Integrate WebSpellChecker products into your web application.

> ⚠ **Do not use 2880 port** when accessing **samples** or other statics files like **wscbundle.js** file as such files will be processed by a selected web server or Java application server. 2880 is an internal port that WSC AppServer listens to and which is designed to process service requests (check spelling, grammar and others).

Option A. Integration Guides

Visit Products and integrations guides to learn more how to get started with WebSpellChecker products, one of them being WProofreader.

WProofreader

WProofreader is multilingual proofreading tool for web applications. It enables grammar and spell check while you type or work with your text in a dedicated dialog. It detects editable fields in focus, be it HTML editable controls or WYSIWYG editors, and enables spelling and grammar checking there. View the full list of supported integrations.

WProofreader benefits:

- Available as a standalone secure solution on your server or as an annual subscription to the service hosted by WebSpellChecker.
- Out of the box offer provides spell check for 17 languages and grammar check for 15 out of 17 default languages;
- All modern web browsers are supported including the latest stable versions of Chrome, Firefox, Safari, MS Edge, and Internet Explorer 11 and mobile web browsers such as Chrome for Android and iOS Safari;
- Over 150 additional languages as well as specialized medical and legal dictionaries are available to extend the default languages set;
- Global custom dictionaries managed by system admins and intended for all WProofreader users;
- Personal user dictionaries for end-users;
- WAI-compliant, up to WCAG 2.1 and Section 508 accessibility standards;
- Operation and installation via HTTPS protocol is supported to fulfill strict security requirements.

With the Cloud-based version of WProofreader you can access and use its features without server-side installation. In the Server-based version, all the application data is securely processed on your own server. Refer to the Get started with WProofreader Server (autoSearch) guide for more details.

WProofreader allows users to see and correct spelling and grammar problems on the fly while they are typing directly in the text input field. Every misspelled word found will be underlined with a red wavy line and every grammar problem detected will be marked with a green one respectively. The user just needs to hover a marked word or phrase and replace it with a proper one from the list of given corrections.

Option B. Demo Samples

Depending on the type of elements, namely HTML native elements or rich text editors where the spelling and grammar check is needed, you may choose and use the demo samples that are shipped with the package.

To access the default samples, go to http(s)://your_host_name:port/wscservice/samples/ where port is the **port** of your web server or Java application server. The default samples contains all the necessary samples and instructions that will help with the integration of the WebSpellChecker products. Moreover, each sample contains detailed development documentation with a list of methods and parameters available.