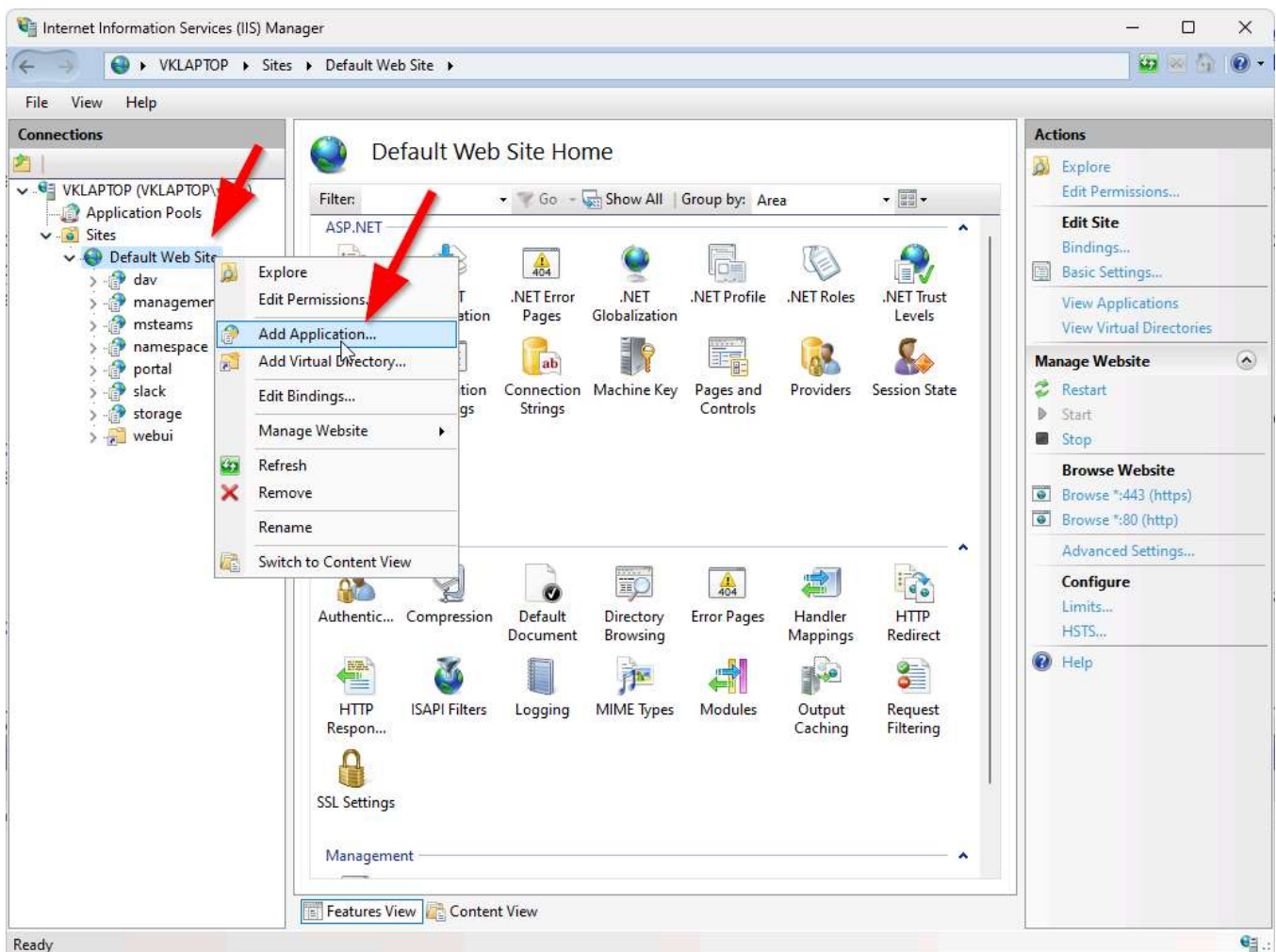


AI Skills Hosting Setup

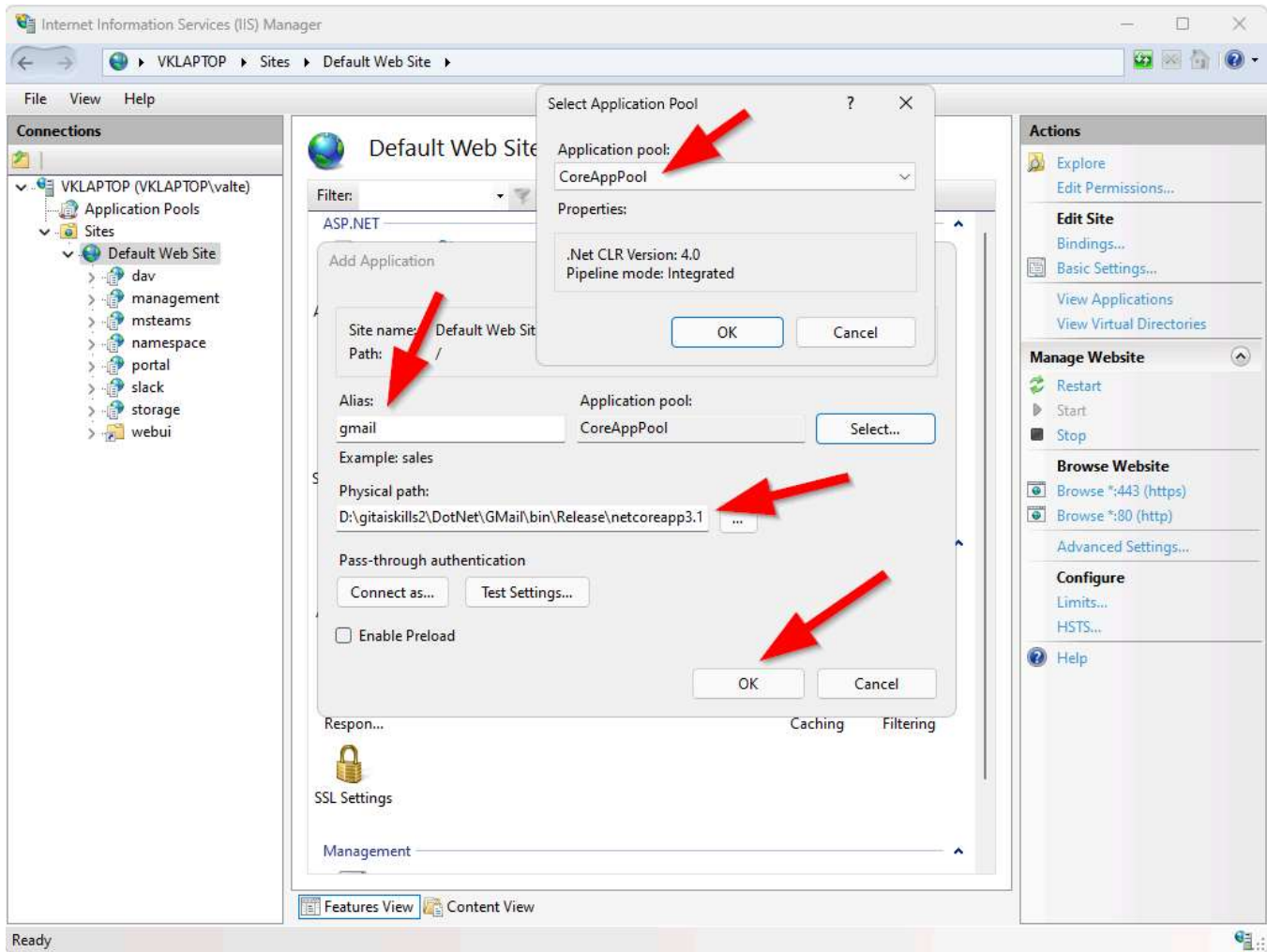
IIS Installation

Custom skills require a Windows Server and Internet Information Services (IIS) with dotnetcore 3.1 Hosting Bundle (dotnet-hosting-3.1.32-win.exe or higher from <https://dotnet.microsoft.com/en-us/download/dotnet/3.1>). The recommended setup is to create a separate web application folder under the Default Web Site node for each custom skill. Each web application folder should have its own application pool so it runs in a separate memory space, and should point to the publishing directory of each Visual Studio project.

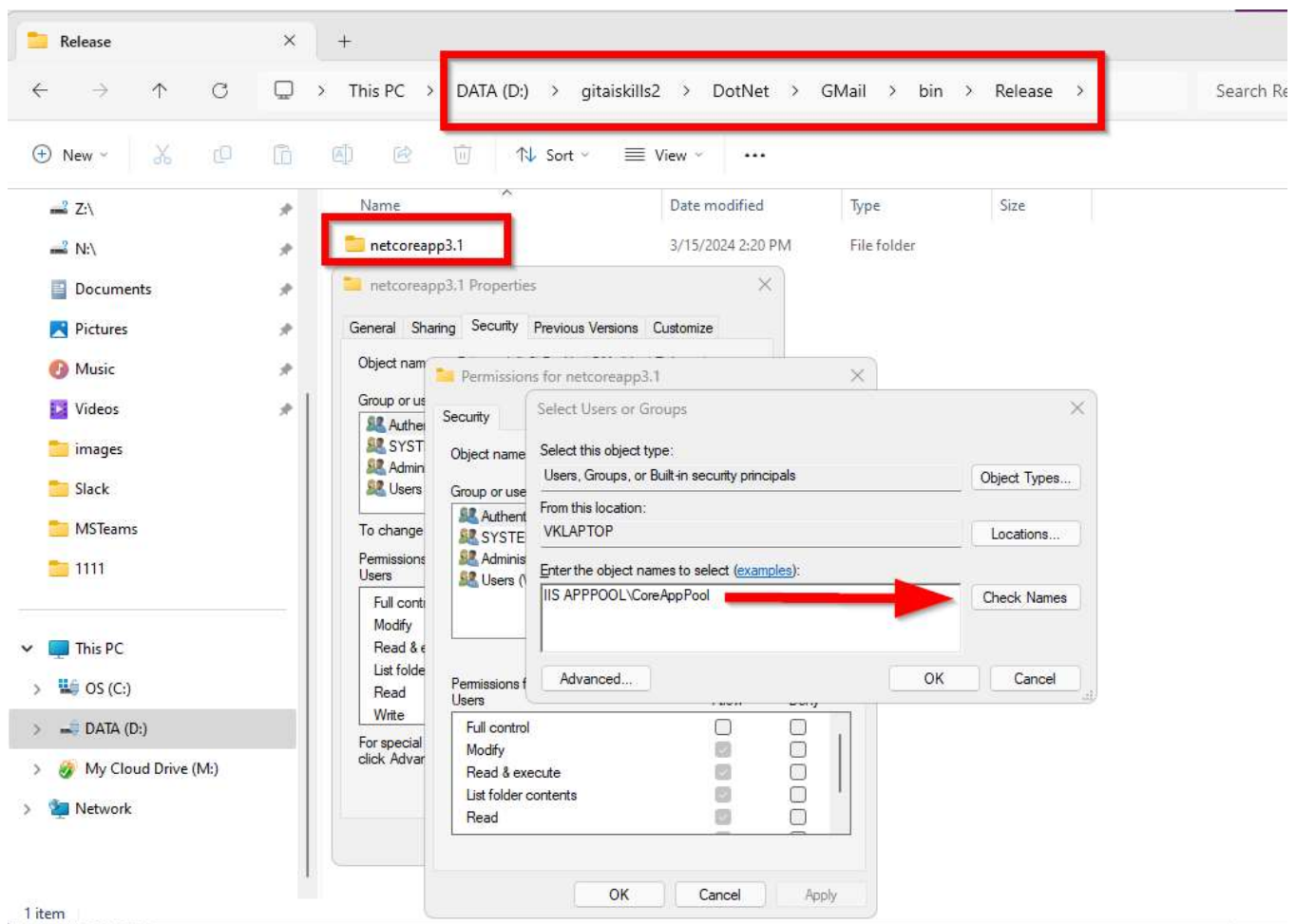


After downloading the source files to a folder on your computer, open the solution (.sln) file with Visual Studio 2022 or higher and compile it. Visual Studio will automatically create a subfolder under the "bin" folder called netcoreapp3.1:

[projectfolder]\bin\Release\netcoreapp3.1 Use the full path similar the one above as the destination path for your IIS web application folder (under Basic Settings...).



Please make sure the IIS application pools and credentials under the web application's basic setting window have the proper NTFS permissions to access the publishing directory. To grant NTFS permissions on a folder to a specific IIS application pool, you can use the following syntax in the NTFS permissions popups to resolve an application pool name: "IIS APPPOOL\[ApplicationPoolNameHere]" (remove the quotes and replace the [ApplicationPoolNameHere] with the actual name.



Testing the Skill Server Setup

If the web application is set up correctly, you should be able to access the following URL from any browser on the server itself: [http://localhost/\[WebAppNameHere\]](http://localhost/[WebAppNameHere]) (Replace *[WebAppNameHere]* with the actual name of your web application. Example: "skill"). The contents of the /Templates/ai-plugin.json, which is an embedded resource in the Visual Studio project is routed from the Controller code of the application, and should be rendered on the page.

To ensure that the routing is working as intended, try to access the following URL from your browser: [http://localhost/\[WebAppNameHere\]/apidefs](http://localhost/[WebAppNameHere]/apidefs) If the above is set up correctly, you should see the available API definitions in YAML format. This page describes the endpoints and parameters that can be accessed via REST (REpresentational State Transfer) requests.

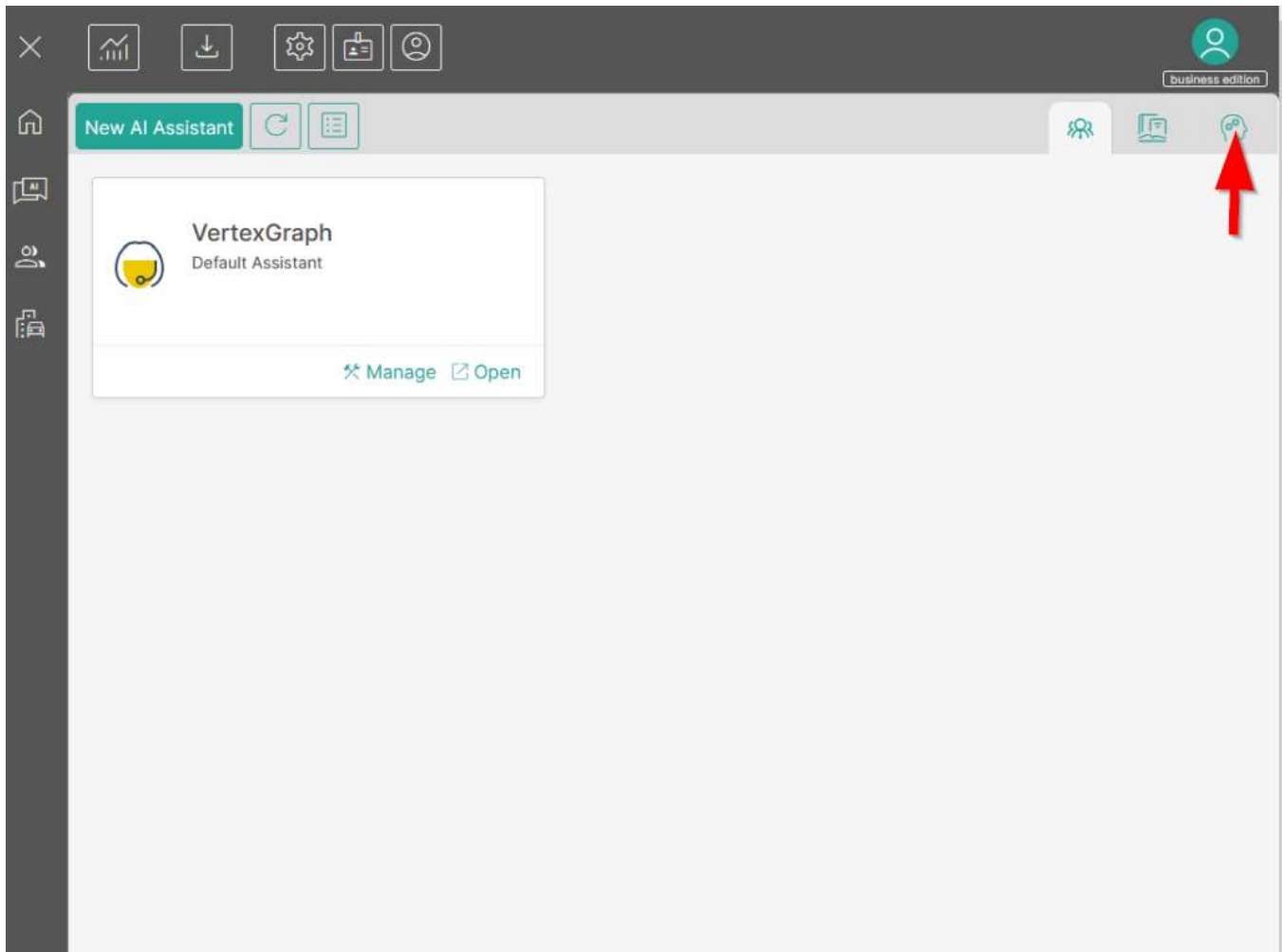
Once you get the above steps working, you can download the Gateway Agent from the VertexGraph.ai website and install it on the same server. The Gateway Agent will allow the skills to be accessible via localhost from the vertexgraph.ai domain. To download the Gateway Agent, set up a New On-Premises Site.

The screenshot shows the 'New On-Premise Site' form in the VertexGraph.ai interface. The form has a title bar with a back arrow and the text 'New On-Premise Site'. Below the title bar, there are two input fields: 'Site Display Name' with the value 'MySkillServer' and 'Description'. A green 'Continue' button is located at the bottom right of the form. The interface includes a top navigation bar with icons for home, download, settings, and user profile, and a left sidebar with icons for home, chat, and user profile. The user profile icon in the top right corner is labeled 'business edition'.

The screenshot shows the 'Download and install Gateway Agent' form in the VertexGraph.ai interface. The form has a title bar with a back arrow and the text 'Download and install Gateway Agent'. Below the title bar, there is a paragraph of text: 'You will need to install a Gateway Agent on box in the site you want to connect. Install the agent on an existing windows file server for better and more functionality.' Below the text, there is a green link 'Download and install Gateway Agent'. At the bottom of the form, there are two buttons: 'I have already installed the Gateway Agent' (green) and 'Back' (white).

Setting up a Custom Skill in VertexGraph

1. Navigate to <https://vertexgraph.ai/portal/>
2. On the left side tool bar select the second option "AI"
3. Now select the custom skills icon on the top right corner



1. Once on the skills section Press "New Custom Skill"
2. Follow the setup until you reach the section where you must enter the OAuth Client ID and Secret
3. Copy and paste the client ID and secret from the Custom App (read the README.md file for each skill for details on how to set up the custom app with the necessary permissions and OAuth Secret keys)
4. Get the Authorization Code and authorize VertexGraph to access your app's account
5. Once completed assign the new skill to any of your assistants to start using it

6. After the setup, open the AI assistant and ask some test prompts. The README.md file for each skill has some test prompts that you can try.