Run/Debug Configuration: PHPUnit on Server

Use this dialog box to configure running unit tests of PHP applications on a remote server.

PHP unit testing requires the following prerequisites to be fulfilled:

■ The PHPUnit ◨ tool should be installed and configured ◨ on your machine.

■ The PEAR folder should be added to the list of the project content roots.

Click here for the description of the options that are common for all run/debug configurations.

The dialog box contains the following tabs and areas:

■ Configuration Tab

■ Test Groups Tab

■ Remote

■ Toolbar

■ Common Options

Configuration Tab

In this tab, specify the unit tests to launch and the command line switches to be passed to the test runner.

Item	Description

Item	Description

Test	In this area, specify the location of tests or the configuration file where they are listed.
	
	■ **All in Directory:** select this option to have all the unit tests in a directory launched.
	
	  In the **Directory** text box, specify the directory to search the unit test in. Type the path to the directory manually or click the **Browse** button 🔲 and select the desired directory in the **Choose Test Directory** dialog box, that opens.
	
	■ **All in File:** select this option to have all the unit tests in a test class or a file launched. In the **File** text box, specify the file to search the class or suite in. Type the path to the file manually or click the **Browse** button 🔲 and select the desired directory in the **Choose Test File** dialog box, that opens.
	
	■ **Class or Suite:** select this option to have all the unit tests in a test class or a test suite launched.
	
	  1. In the **File** text box, specify the file to search the class or suite in. Type the path to the file manually or click the **Browse** button 🔲 and select the desired directory in the **Choose Test File** dialog box, that opens.
	
	  2. In the **Class** text box, specify the desired class. Type the class name manually or click the **Browse** button 🔲 and select the desired class in the tree view, that opens.
	
	■ **Method:** select this option to have a specific test method launched.
	
	  1. In the **File** text box, specify the file to search for the test method in. Type the file name manually or click the **Browse** button 🔲 and select the desired file in the tree view, that opens.
	
	  2. In the **Method** text box, specify the desired method.
	
	■ **XML file:** select this option to have test runner execute the tests from a dedicated XML file. Specify the XML file to use in the **Use XML configuration file** text box.

Custom working directory	In this text box, specify the location of the files that are outside the folder with tests and are referenced in your tests through relative paths. Type the path manually or click the **Browse** button 🔲 and select the desired folder in the dialog that opens. This settings does not block the test execution because the location of tests is always specified through a full path to the corresponding files and/or directories.

Test Groups Tab

In this tab, appoint the groups of tests to execute when the tests are run according to the current run configuration.

Grouping tests ⧉ is helpful, for example, to distinguish between tests to run in a production environment from those to run in your development environment. You just need to create two groups and then include or exclude them depending on the current environment.

To attach a test to a group, tag it with the @group ⊡ annotation in the format:

```
/**
    @group <group specification>
*/
```

To enable filtering tests based on their authors, tag the tests with the @author ⊡ annotation.

Item	Description
Include/Exclude groups	Select this check box to enable configuring execution of test groups.
Group Name	In this read-only list, select the test group to be involved or skipped in testing.
Include	Select this check box to have the tests from the selected test group executed.
Exclude	Select this check box to have the tests from the selected test group skipped.

Remote

In this tab, configure deployment of tests to a remote server.

Item	Description
Debug server	In this field, specify the configuration through which the debugging engine will interact with the server through HTTP. Actually, a configuration consist of a URL address of the target environment, the debugger type, and possibly mappings between folders in your project and their correspondences on the server. Choose a configuration from the drop-down list or click the **Browse** button ⊡ and define a new configuration in the Servers dialog box that opens.
Upload test directory before run	■ When this check box is selected, IntelliJ IDEA automatically uploads tests to the *default* server according to the deployment options. ■ When the check box is cleared, you have to upload tests manually, see Uploading and Downloading Files. The check box is by default selected.
Remove test files after run	■ When this check box is selected, IntelliJ IDEA automatically removes the executed tests from the server upon execution. ■ When the check box is cleared, tests remain on the server after execution and you will have to download them manually, see Uploading and Downloading Files.
Show transfer logs	Select this check box to have test deployment logged.

Toolbar

Item	Shortcut	Description
✚	Alt+Insert	Click this button to add new configuration to the list.
━	Alt+Delete	Click this button to remove the selected configuration from the list.
▯	Ctrl+D	Click this button to create a copy of the selected configuration.
🔧	Edit defaults	Click this button to edit the default configuration templates. The defaults are used for the newly created configurations.
⬆ or ⬇	Alt+Up or Alt+Down	Use these buttons to move the selected configuration or group of configurations (folder) up and down in the list. The order of configurations or folders in the list defines the order, in which configurations appear in the **Run/Debug** drop-down list on the main toolbar.
📁		Use this button to create a new folder. If one or more run/debug configurations have the focus, then the selected run/debug configurations are automatically moved to the newly created folder. If only a category has the focus, then an empty folder is created. Move run/debug configurations to a folder using drag-and-drop, or ⬆ ⬇ buttons.

Common options

Item	Description
Name	In this text box, specify the name of the current run/debug configuration. This field does not appear for the default run/debug configurations.
Defaults	This node in the left-hand pane of the dialog box contains the default run/debug configuration settings. Select the desired configuration to change its default settings in the right-hand pane. The defaults are applied to all newly created run/debug configurations.
Share	Select this check box to make the run/debug configuration available to other team members. If the directory-based project format is used, the settings for a run/debug configuration are stored in a separate xml file in the `.idea\runConfigurations` folder if the run/debug configuration is shared and in the `.idea\workspace.xml` file otherwise. If the file-based format is used, the settings are stored in the `.ipr` file for shared configurations or in the `.iws` file for the ones that are not shared. This check box is not available when editing the run/debug configuration defaults.

Item	Description
Before launch	Specify which tasks should be carried out before starting the run/debug configuration. The specified tasks are performed in the order that they appear in the list.

Item	Keyboard shortcut	Description
✚	Alt+Insert	Click this icon to add a task to the list. Select the task to be added: ■ **Run External tool.** Select this option to run an application which is external to IntelliJ IDEA. In the dialog that opens, select the application or applications that should be run. If the necessary application is not defined in IntelliJ IDEA yet, add its definition. For more information, see Configuring Third-Party Tools and External Tools. ■ **Build Artifacts.** Select this option to have an artifact or artifacts built. In the dialog that opens, select the artifact or artifacts that should be built. See also, Configuring Artifacts. ■ **Run Ant target.** Select this option to have an Ant target run. In the dialog that opens, select the target to be run. For more information, see Ant. ■ **Generate CoffeeScript Source Maps.** Select this option to have the source maps for your CoffeeScript sources generated. In the dialog that opens, specify where your CoffeeScript source files are located. For more information, see CoffeeScript Support. ■ **Run Maven Goal.** Select this option to have a Maven goal run. In the dialog that opens, select the goal to be run. For more information, see Maven. ■ **Run Remote External tool**: Add a remote SSH external tool. Refer to the section Remote SSH External Tools for details.
─	Alt+Delete	Click this icon to remove the selected task from the list.
✎	Enter	Click this icon to edit the selected task. Make the necessary changes in the dialog that opens.
⬆	Alt+Up	Click this icon to move the selected task one line up in the list.
⬇	Alt+Down	Click this icon to move the selected task one line down in the list.

Item	Item	Keyboard shortcut	Description
	Show this page		Select this check box to have the run/debug configuration settings shown prior to actually starting the run/debug configuration.

See Also

Concepts:

- Run/Debug Configuration

Procedures:

- Creating Run/Debug Configuration for Tests
- Testing
- Testing PHP Applications
- Configuring Content Roots

Reference:

- Sources Tab

Web Resources:

- Developer Community ⧉