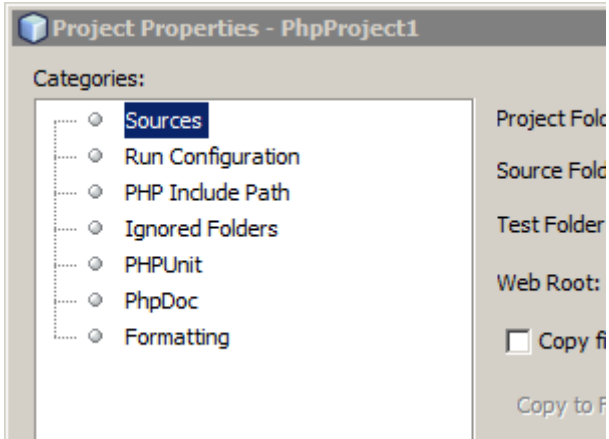


System	Windows XP SP3 version 5.1 running on x86; Cp1252; en_US (nb)
Product Version	NetBeans IDE 7.0.1 RC1 (Build 201106222100)
Java	1.7.0; Java HotSpot(TM) Client VM 21.0-b17
Apache	2.2.18
MySQL	5.5.13
PHP	5.3.6
Symfony	1.4.12
Zend Framework	1.11.8
Test Specification	TS 70 PHP Sanity
RESULT	GO Total Test Cases: 70 OK: 50 VARIANCE: 4 FAIL: 11 Not Performed: 5
Author	Siddharth Kumar Singh
Date of Completion	12 July 2011

TS 69 PHP Sanity

Test	Result	Remarks
Test suite: Create a new PHP project		
1. Create an empty Project with default settings	OK	
2. Create empty Project with the given name	OK	
3. Create empty Project with incorrect name	OK	
4. Create empty Project and put metadata into a separate directory	OK	
5. Create empty Project with renamed index.php file	OK	
6. Create empty Project and choose copy location	OK	
7. Create Symfony project	NOT PERFORMED	
8. Create Zend project	NOT PERFORMED	
9. Framework enhancements	NOT PERFORMED	
Test Suite: Create Sample Application		
1. Create Air Alliance Sample	OK	
2. Create Jobeet Sample	NOT PERFORMED	

Test Suite: Check Properties of PHP Project		
1. Check properties of PHP Project with default settings	VARIANCE	<p>EXPECTED RESULT: Project Properties window appears. There are six Categories: Sources, Run Configuration, PHP Include Path, JavaScript Libraries, Debug, Formatting.</p> <p>ACTUAL RESULT: Project Properties window appears. There are seven Categories: Sources, Run Configuration, PHP Include Path, Ignored Folders, PHPUnit, PhpDoc, Formatting.</p> 
Run file without project		
1. Test running file without project	OK	
Debug file without project		
1. Test debugging file without project.	OK	
Test suite: Code Completion		
1. Create a PHP source file	OK	
2. Verify automatic code completion invocation	OK	
3. Verify local variable code completion	OK	
4. Verify global variable code completion	VARIANCE	<p>EXPECTED RESULT: Code completion appears and the variable name is among suggestions.</p> <p>ACTUAL RESULT: Code completion works in the main body of the php file but not inside the body of a function</p>

5. Verify variable from included file code completion	VARIANCE	EXPECTED RESULT: Code completion appears and the variable name is among suggestions. ACTUAL RESULT: Code completion works in the main body of the php file but not inside the body of a function
6. Verify variable from required file code completion	VARIANCE	EXPECTED RESULT: Code completion appears and the variable name is among suggestions. ACTUAL RESULT: Code completion works in the main body of the php file but not inside the body of a function
7. Verify code completion inside the identifier	OK	
8. Verify documentation hints for built-in identifiers	FAIL	EXPECTED RESULT: In the details code completion view, a description of the identifier appears. ACTUAL RESULT: No description appears though the keyword does show up in autocomplete.
9. Verify documentation hints for keywords	FAIL	EXPECTED RESULT: In the details code completion view, a description of the keyword appears. ACTUAL RESULT: No description appears though the keyword does show up in autocomplete.
10. Verify keywords code completion	OK	
11. Verify code completion with a single option	OK	
12. Verify JavaDoc window	OK	
13. Verify code completion after EXTENDS	OK	
14. Verify code completion in // comments	OK	
15. Verify code completion in /* */ comments	FAIL	EXPECTED RESULT: Code completion doesn't appear. It shouldn't be available in comments. ACTUAL RESULT: Code completion appears
16. Verify code completion in /** */ comments	OK	
Test suite: Code Folding		
1. Create a PHP web page	OK	

2. Verify code fold tooltip	FAIL	EXPECTED RESULT: Full content of folded code should appear as a tooltip. ACTUAL RESULT: Code folding does not work for PHP code embedded inside HTML
3. Expand code fold by double click	FAIL	EXPECTED RESULT: Code fold is expanded. ACTUAL RESULT: Code folding does not work for PHP code embedded inside HTML
4. Recurrent Code folding	FAIL	EXPECTED RESULT: When expanding the main code fold back, other code fold should be in the same states as they were before collapsing the main code fold. ACTUAL RESULT: Code folding does not work for PHP code embedded inside HTML
5. Repeat all previous testcases for HTML folding	OK	
6. PHP functions code folding	FAIL	EXPECTED RESULT: The code fold should exist and work properly. The function title should be displayed outside of code fold. ACTUAL RESULT: Code folding does not work for PHP code embedded inside HTML
7. PHP <code>/**/</code> comments code folding	FAIL	EXPECTED RESULT: The code fold should exist and work properly. ACTUAL RESULT: Code folding does not work for PHP code embedded inside HTML
8. PHP <code>//</code> comments code folding	OK	
9. Copy/paste code folds	FAIL	EXPECTED RESULT: Code is pasted properly and is unfolded. ACTUAL RESULT: Code folding does not work for PHP code embedded inside HTML
10. Verify code folding from the main menu	OK	
11. Verify code folding via keyboard shortcuts	OK	
Test suite: Mark Occurrences		
1. Create a PHP web page	OK	
2. Verify local variable occurrences marking	OK	

3. Verify function occurrences marking	OK	
4. Verify parameter occurrences marking	OK	
Test suite: HTML Palette		
1. Create a PHP web page	OK	
2. DnD Table HTML component to HTML code	OK	
3. DnD Table HTML component to PHP code	OK	
4. DnD Table HTML component outside of editor view	OK	
5. DnD big Table HTML component to HTML code	FAIL	<p>EXPECTED RESULT: After reasonable time (few seconds), HTML tags added to DnD position. They correspond to the visual representation of Table HTML component in the "Insert Table" dialog (with made changes).</p> <p>ACTUAL RESULT: The IDE becomes unresponsive for some time post which it renders strangely</p>
6. DnD Table HTML component with given parameters to HTML code	OK	
7. Add Table HTML component with given parameters to HTML code using double-click	OK	
8. Undo adding Table HTML component to HTML code	OK	
9. DnD Form component with given parameters to HTML code	OK	
10. DnD other HTML components with given parameters to HTML code	OK	

Test suite: Formatting		
1. Create a PHP web page	OK	
2. Format default code of PHP web page	OK	
3. Undo formatting of PHP web page	OK	
4. Create a PHP file	OK	
5. Format default code of PHP file	OK	
6. Undo formatting of PHP file	OK	
7. Formatting of folded code	OK	
Test suite: Navigation		
1. Create a PHP web page	OK	
2. Navigate to a specified line	OK	
3. Navigate to an invalid line	OK	
Test suite: Type Hierarchy View		
1. Type Hierarchy View	OK	
Test suite: Aliases		
1. Aliases	OK	
Test suite: Refactoring		
1. Rename refactor a class method	FAIL	<p>EXPECTED RESULT: bar(\$barStr) is renamed to netCat(\$barStr). Foo::work() now calls netCat() instead of bar(). Method call in index.php is also renamed.</p> <p>ACTUAL RESULT: Only bar(\$barStr) is renamed. Other calls are left untouched.</p>
2. Safely delete a class	OK	
Test suite: PHPDoc		
1. Generate documentation for a project	NOT PERFORMED	