

# PhpStormとPHPUnitを 連携してユニットテスト 作成を楽にする

2019/1/26 PHPカンファレンス仙台2019

田中孝治 @tenkoma

最近、ユニットテスト  
書けてますか？

テスト書いて実行の繰り返し  
が面倒だと思ったことは  
ありますか？

PhpStormとPHPUnitを連携  
するとテスト作成が楽になり、  
コード理解もしやすくなります  
試してみてください

# ちなみに PhpStormを使っている方



すでにPHPUnitと連携して  
テスト書いてるよという方



# 話すこと

- 🙋 PHPUnitとは、PhpStorm連携とは
- 🙋 連携できてなかった開発環境
- 🙋 連携の設定
- 🙋 ユニットテスト作成がどう楽になるか(デモあり)
- ~~✖ ユニットテストの基本的な書き方~~
- ~~✖ PhpStormの基本的な使い方~~

# 自己紹介

- 田中孝治(Tanaka Koji)
- @tenkoma (GitHub, Twitter…)
- BASE株式会社 Product Division  
Backend Engineer (2017/9～)
  - PHP/CakePHP バージョンアップ・モバイルアプリバックエンド開発
  - ユニットテストのしやすさ、安定性の改善
- Contributed to CakePHP





PHPUnitとは  
PhpStorm連携とは

# PHPUnit

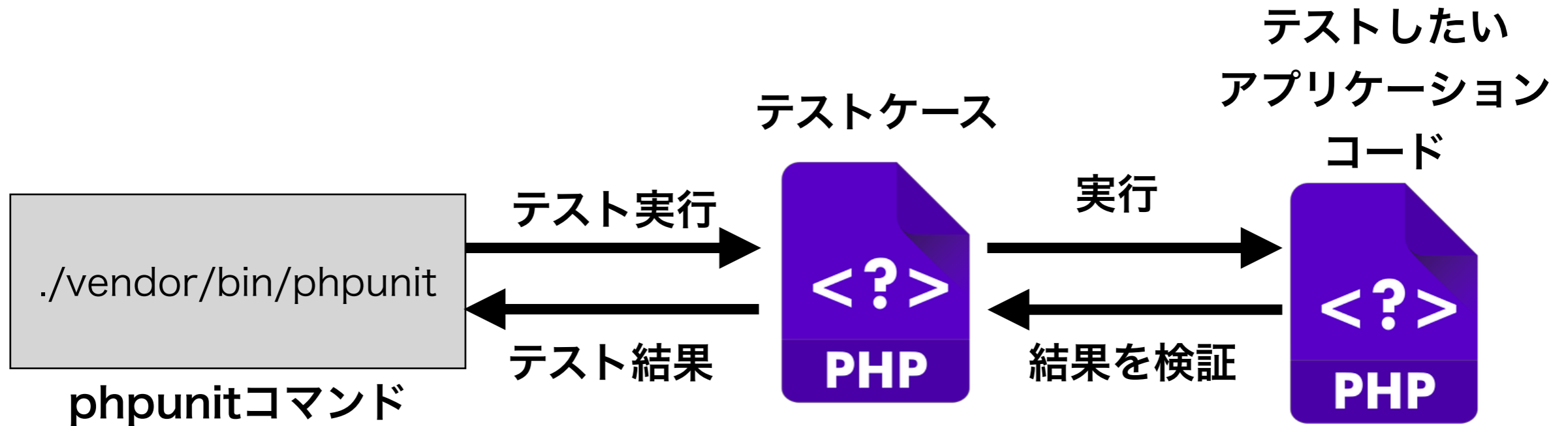
- PHPコードのユニットテストを書くためのフレームワーク
- 多くのPHPライブラリで使われている
- Webアプリケーションのテスト基盤としても使われる

※大規模PHPプロジェクトでPHPUnitを3世代アップグレードするためにやったこと

<https://speakerdeck.com/dqneo/phpunit-upgrade-story?slide=18>

を参考にしました

# PHPUnit



すべてパス

```
PHPUnit 6.5.13 by Sebastian Bergmann and contributors.  
..... 42 / 42 (100%)  
Time: 13.95 seconds, Memory: 26.00MB  
OK (42 tests, 101 assertions)
```

1つ以上失敗

```
PHPUnit 6.5.13 by Sebastian Bergmann and contributors.  
..F. 4 / 4 (100%)  
Time: 1.25 seconds, Memory: 16.00MB  
There was 1 failure:  
1) App\Test\TestCase\Controller\UsersControllerTest::testLoginSucceed  
Failed asserting that response has header 'Location'.  
/var/www/html/vendor/cakephp/cakephp/src/TestSuite/IntegrationTestTrait.php:842  
/var/www/html/tests/Testcase/Controller/UsersControllerTest.php:59  
FAILURES!  
Tests: 4, Assertions: 8, Failures: 1.
```

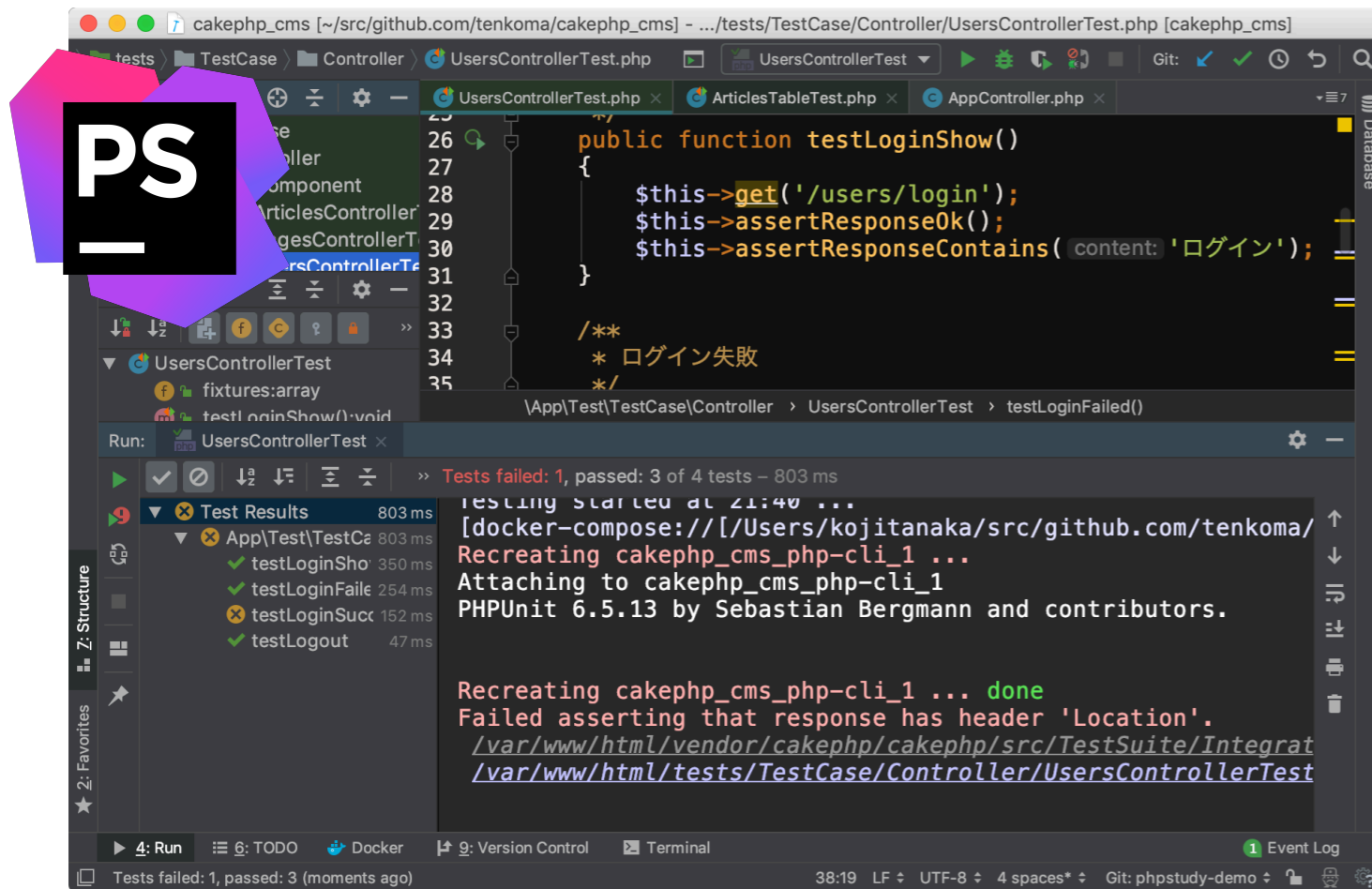
アイコン画像

<https://pixabay.com/ja/users/mtzd-1593970/>

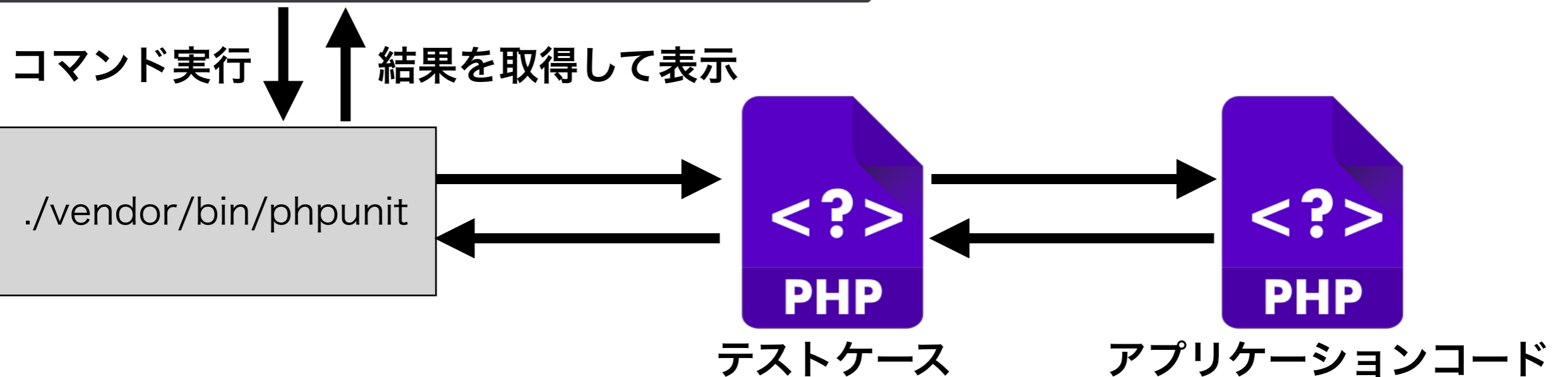
# PhpStorm

- PhpStorm: PHPアプリケーションのための統合開発環境 (IDE)
- エディタ・デバッガ連携・静的解析・バージョン管理etc...

# PhpStormと連携

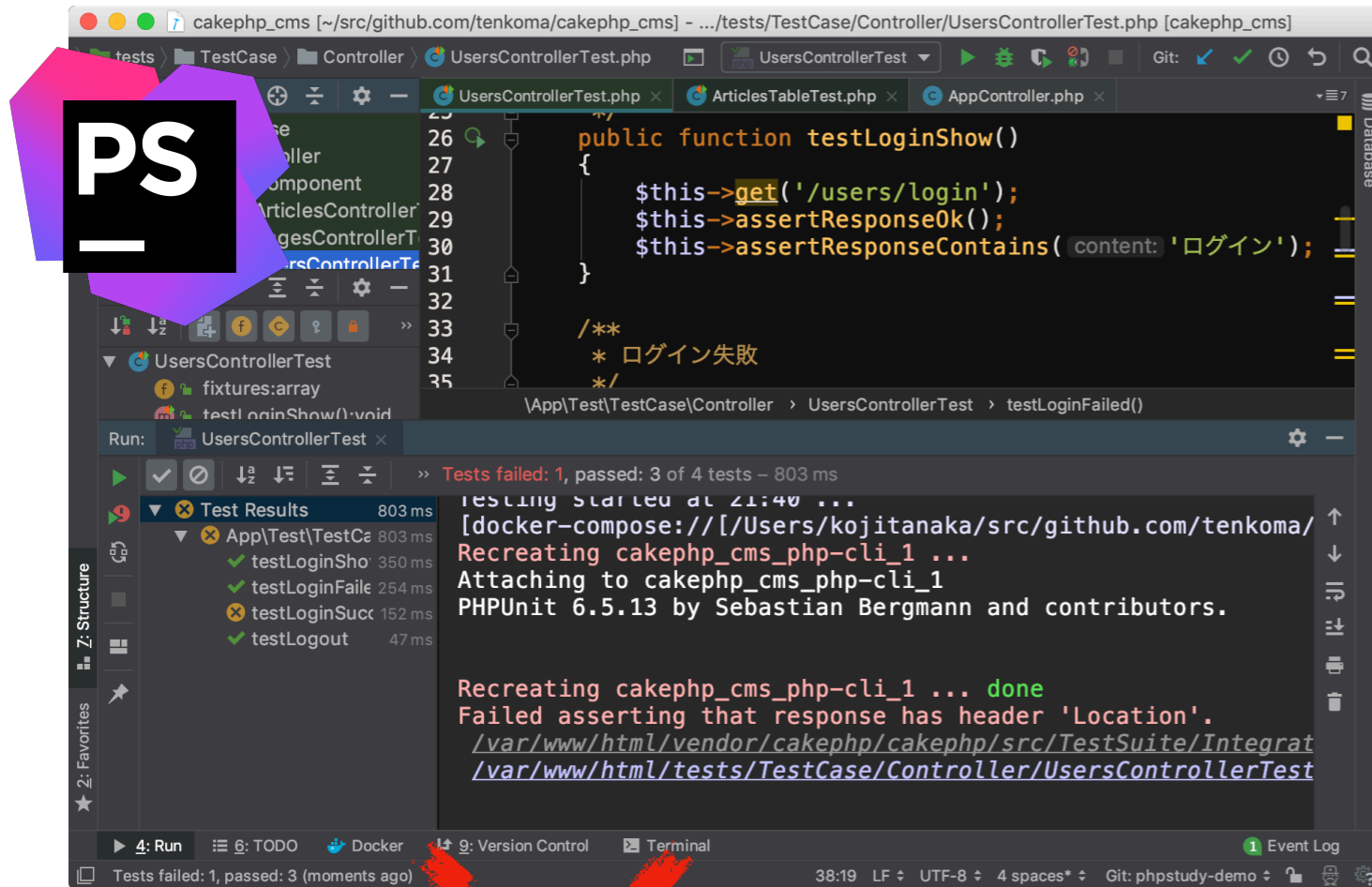


連携することで、  
PhpStorm からテスト  
実行できる



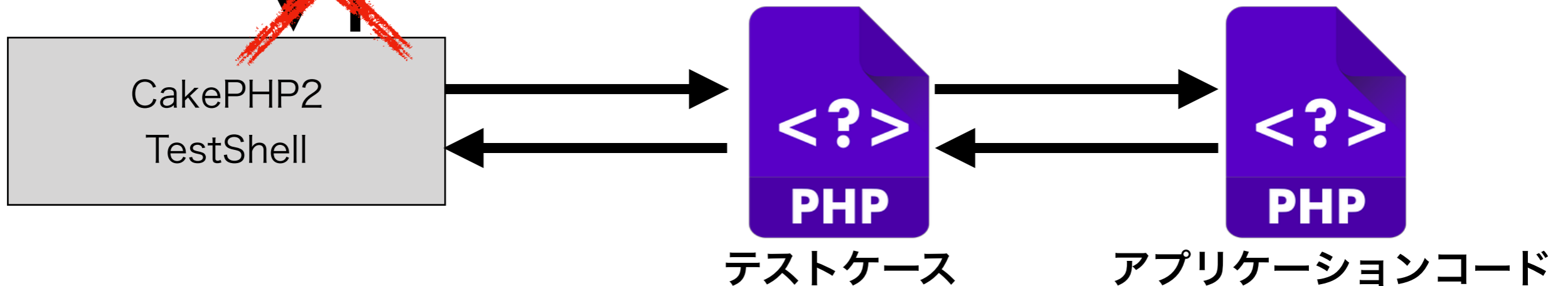
連携できてなかった開発環境  
(CakePHP2アプリ)

# CakePHP2固有の事情



- テスト実行コマンドがカスタマイズされている

コマンド実行 ↓ ↑ 結果を取得して表示



# エディタでコード変更

ターミナルに切り替えてテスト実行

…繰り返し…

The image shows a screenshot of an IDE (likely PhpStorm) with a code editor and a terminal window. The code editor displays the following PHP code for a test case:

```
21 ];
22
23 /**
24  * ログインページが表示される
25  */
26 public function testLoginShow()
27 {
28     $this->get('/users/login');
29     $this->assertResponseOk();
30     $this->assertResponseContains('content: 'ログイン');
31 }
32
33 /**
34  * ログイン失敗
35  */
36 // public function testLoginFailed()
37 // {
38 //     $this->enableCsrfToken();
```

The terminal window shows the following output:

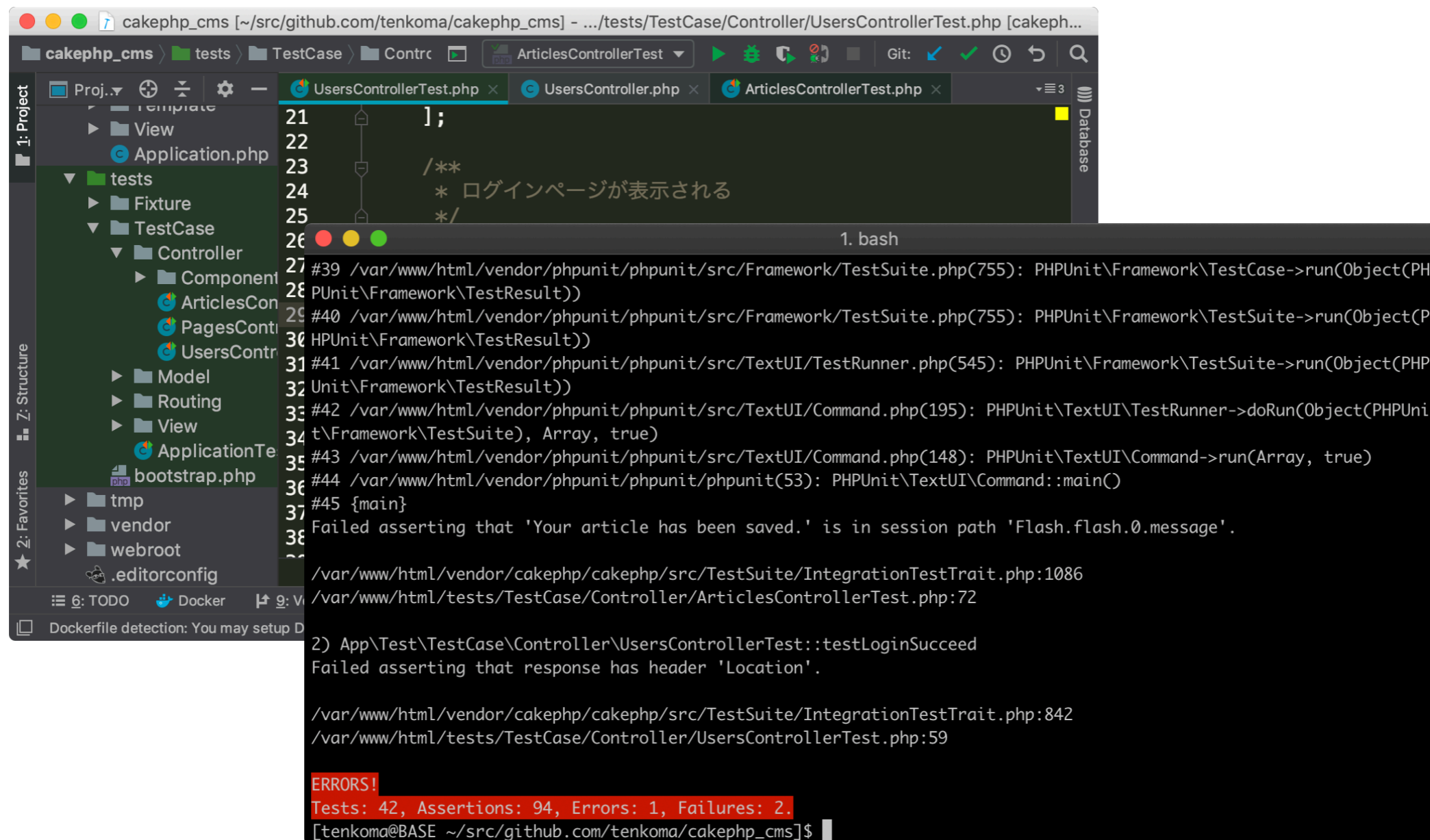
```
2) App\Test\TestCase\Controller\UsersControllerTest::testLoginSucceed
Failed asserting that response has header 'Location'.

/var/www/html/vendor/cakephp/cakephp/src/TestSuite/IntegrationTestTrait.php:842
/var/www/html/tests/TestCase/Controller/UsersControllerTest.php:59

ERRORS!
Tests: 42, Assertions: 94, Errors: 1, Failures: 2.
[tenkoma@BASE ~/src/github.com/tenkoma/cakephp_cms]$
```



# エディタでコード変更 ターミナルに切り替えてテスト実行 …繰り返し…

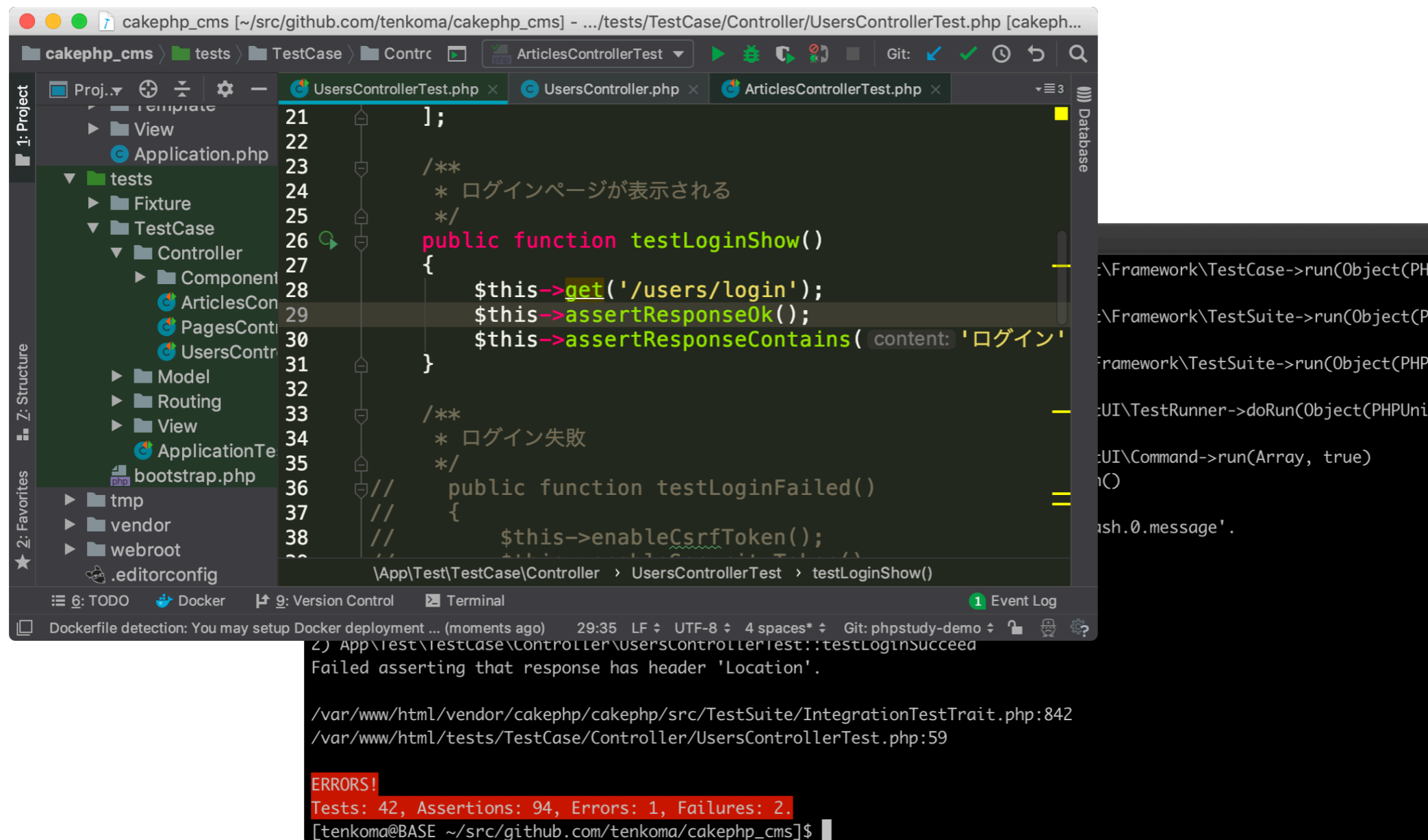


The screenshot shows an IDE window with a file explorer on the left displaying a project structure for 'cakephp\_cms'. The main editor shows a PHP file 'ArticlesControllerTest.php' with a comment in Japanese: 'ログインページが表示される'. A terminal window is overlaid on the bottom right, showing the execution of PHPUnit tests. The terminal output includes several lines of PHP error messages and a summary of test results: 'Tests: 42, Assertions: 94, Errors: 1, Failures: 2.' The terminal prompt is '[tenkoma@BASE ~/src/github.com/tenkoma/cakephp\_cms]\$'.

```
cakephp_cms [~/src/github.com/tenkoma/cakephp_cms] - .../tests/TestCase/Controller/UsersControllerTest.php [cakeph...
cakephp_cms > tests > TestCase > Contrc > ArticlesControllerTest
ArticlesControllerTest
Git: [check] [refresh] [undo] [redo] [search]
1. Project
  cakephp_cms
  - Template
  - View
  - Application.php
  - tests
    - Fixture
    - TestCase
      - Controller
        - Component
        - ArticlesCon
        - PagesConti
        - UsersContr
      - Model
      - Routing
      - View
    - ApplicationTe
    - bootstrap.php
  - tmp
  - vendor
  - webroot
  - .editorconfig
2. Favorites
3. Structure
4. Docker
5. Dockerfile detection: You may setup D

21 ];
22
23 /**
24  * ログインページが表示される
25  */
26
27 #39 /var/www/html/vendor/phpunit/phpunit/src/Framework/TestSuite.php(755): PHPUnit\Framework\TestCase->run(Object(PH
28 PHPUnit\Framework\TestResult))
29 #40 /var/www/html/vendor/phpunit/phpunit/src/Framework/TestSuite.php(755): PHPUnit\Framework\TestSuite->run(Object(P
30 PHPUnit\Framework\TestResult))
31 #41 /var/www/html/vendor/phpunit/phpunit/src/TextUI/TestRunner.php(545): PHPUnit\Framework\TestSuite->run(Object(PHP
32 Unit\Framework\TestResult))
33 #42 /var/www/html/vendor/phpunit/phpunit/src/TextUI/Command.php(195): PHPUnit\TextUI\TestRunner->doRun(Object(PHPUni
34 t\Framework\TestSuite), Array, true)
35 #43 /var/www/html/vendor/phpunit/phpunit/src/TextUI/Command.php(148): PHPUnit\TextUI\Command->run(Array, true)
36 #44 /var/www/html/vendor/phpunit/phpunit/phpunit(53): PHPUnit\TextUI\Command::main()
37 #45 {main}
Failed asserting that 'Your article has been saved.' is in session path 'Flash.flash.0.message'.
/var/www/html/vendor/cakephp/cakephp/src/TestSuite/IntegrationTestTrait.php:1086
/var/www/html/tests/TestCase/Controller/ArticlesControllerTest.php:72
2) App\Test\TestCase\Controller\UsersControllerTest::testLoginSucceed
Failed asserting that response has header 'Location'.
/var/www/html/vendor/cakephp/cakephp/src/TestSuite/IntegrationTestTrait.php:842
/var/www/html/tests/TestCase/Controller/UsersControllerTest.php:59
ERRORS!
Tests: 42, Assertions: 94, Errors: 1, Failures: 2.
[tenkoma@BASE ~/src/github.com/tenkoma/cakephp_cms]$
```

# エディタでコード変更 ターミナルに切り替えてテスト実行 …繰り返し…



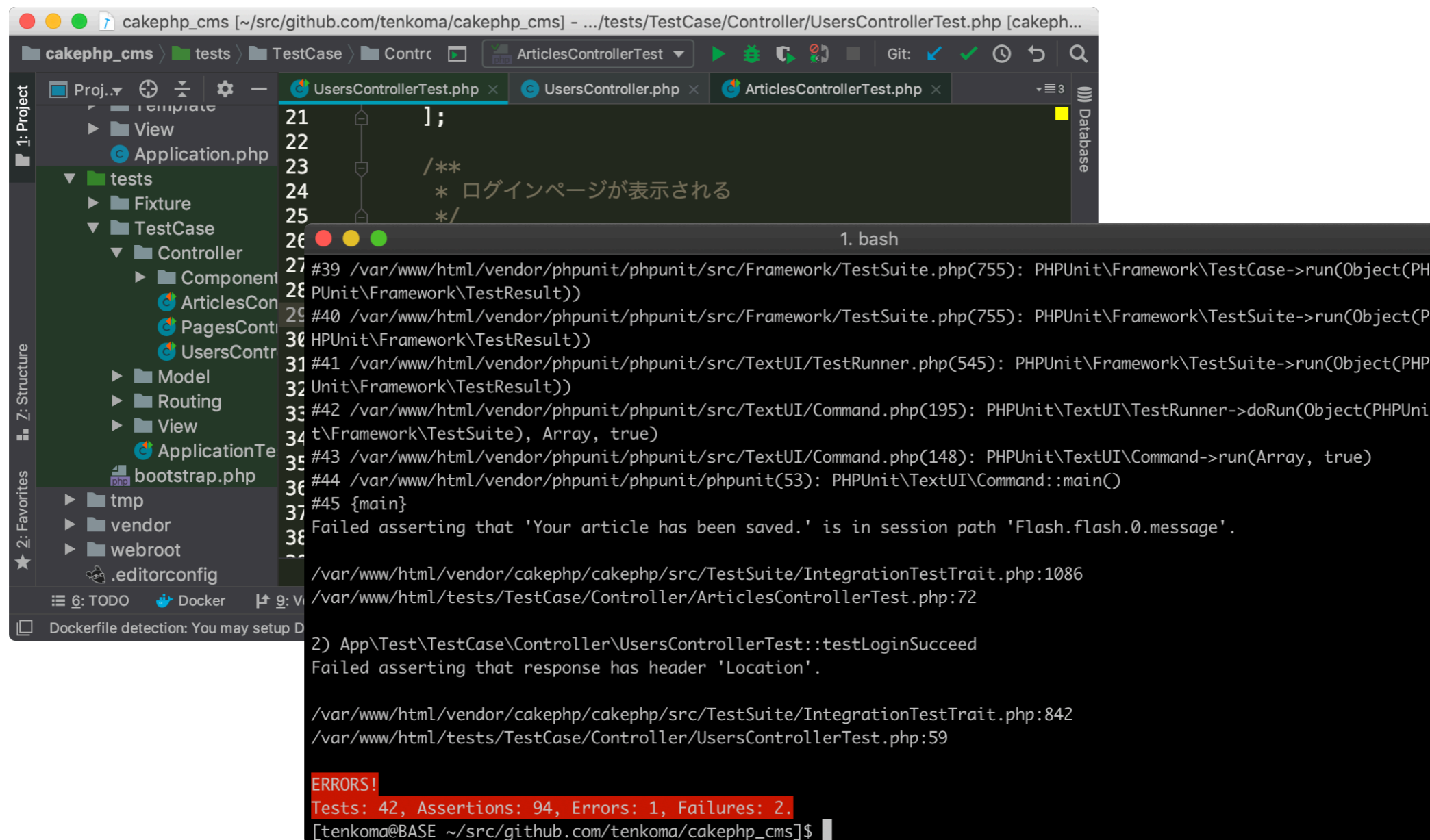
The image shows a screenshot of an IDE (likely PhpStorm) with a terminal window open. The IDE window displays the code for a test case, `UsersControllerTest.php`, with the following content:

```
21 ];  
22  
23 /**  
24  * ログインページが表示される  
25  */  
26 public function testLoginShow()  
27 {  
28     $this->get('/users/login');  
29     $this->assertResponseOk();  
30     $this->assertResponseContains('content: 'ログイン'  
31 }  
32  
33 /**  
34  * ログイン失敗  
35  */  
36 // public function testLoginFailed()  
37 // {  
38 //     $this->enableCsrfToken();  
39 // }
```

The terminal window shows the output of the test execution, including the following error message:

```
2) App\Test\TestCase\Controller\UsersControllerTest::testLoginSucceed  
Failed asserting that response has header 'Location'.  
  
/var/www/html/vendor/cakephp/cakephp/src/TestSuite/IntegrationTestTrait.php:842  
/var/www/html/tests/TestCase/Controller/UsersControllerTest.php:59  
  
ERRORS!  
Tests: 42, Assertions: 94, Errors: 1, Failures: 2.  
[tenkoma@BASE ~/src/github.com/tenkoma/cakephp_cms]$
```

# エディタでコード変更 ターミナルに切り替えてテスト実行 …繰り返し…



The screenshot shows an IDE window with a file explorer on the left and a code editor in the center. The file explorer shows a project structure with folders like 'tests', 'Controller', and 'View'. The code editor shows PHP code for a test case. A terminal window is open in the foreground, displaying the output of a PHPUnit test run. The terminal output shows the test suite running, with a failure message: 'Failed asserting that 'Your article has been saved.' is in session path 'Flash.flash.0.message''. The terminal also shows the test results: 'Tests: 42, Assertions: 94, Errors: 1, Failures: 2.'

```
cakephp_cms [~/src/github.com/tenkoma/cakephp_cms] - .../tests/TestCase/Controller/UsersControllerTest.php [cakeph...
cakephp_cms > tests > TestCase > Contrc > ArticlesControllerTest
UsersControllerTest.php x UsersController.php x ArticlesControllerTest.php x
21 ];
22
23 /**
24  * ログインページが表示される
25  */
26
27 #39 /var/www/html/vendor/phpunit/phpunit/src/Framework/TestSuite.php(755): PHPUnit\Framework\TestCase->run(Object(PH
28 PHPUnit\Framework\TestResult))
29 #40 /var/www/html/vendor/phpunit/phpunit/src/Framework/TestSuite.php(755): PHPUnit\Framework\TestSuite->run(Object(P
30 PHPUnit\Framework\TestResult))
31 #41 /var/www/html/vendor/phpunit/phpunit/src/TextUI/TestRunner.php(545): PHPUnit\Framework\TestSuite->run(Object(PHP
32 Unit\Framework\TestResult))
33 #42 /var/www/html/vendor/phpunit/phpunit/src/TextUI/Command.php(195): PHPUnit\TextUI\TestRunner->doRun(Object(PHPUni
34 t\Framework\TestSuite), Array, true)
35 #43 /var/www/html/vendor/phpunit/phpunit/src/TextUI/Command.php(148): PHPUnit\TextUI\Command->run(Array, true)
36 #44 /var/www/html/vendor/phpunit/phpunit/phpunit(53): PHPUnit\TextUI\Command::main()
37 #45 {main}
Failed asserting that 'Your article has been saved.' is in session path 'Flash.flash.0.message'.
/var/www/html/vendor/cakephp/cakephp/src/TestSuite/IntegrationTestTrait.php:1086
/var/www/html/tests/TestCase/Controller/ArticlesControllerTest.php:72
2) App\Test\TestCase\Controller\UsersControllerTest::testLoginSucceed
Failed asserting that response has header 'Location'.
/var/www/html/vendor/cakephp/cakephp/src/TestSuite/IntegrationTestTrait.php:842
/var/www/html/tests/TestCase/Controller/UsersControllerTest.php:59
ERRORS!
Tests: 42, Assertions: 94, Errors: 1, Failures: 2.
[tenkoma@BASE ~/src/github.com/tenkoma/cakephp_cms]$
```

2018-12-03

# CakePHP2 アプリでも PHPUnit と PhpStormを連携させる

CakePHP Advent Calendar 2018 - Qiita 3日目の記事になります。

去年のCakePHPアドベントカレンダーではCakePHP 3のチュートリアルにユニットテストを追加する話を書きましたが、今年はCake 2のテストの話を書きます。

## CakePHP2 アプリは PhpStorm の恩恵が受けにくい

プロダクト開発でCakePHP2 を使ってアプリを構築していると、PhpStorm 機能の恩恵を受けられない以下のような難点があります。

- DBのデータを取り出すと配列で、コード補完ができない
- 名前空間対応してないので、コード補完ができない
- ユニットテスト実行が、`phpunit` コマンドではなく、独自のコマンド `Console/cake test`

少しPHPコードを追加したら  
連携できたのでブログ記事書いた

<https://tenkoma.hatenablog.com/entry/2018/12/03/113000>

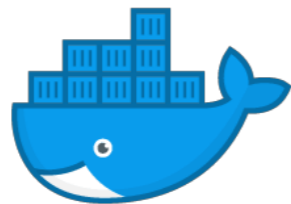
“CakePHP2でアプリ開発してる方は参考に

But this means you can't integrate it with tooling like IDE/PhpStorm which expect to run

# 連携の設定

# 説明用開発環境

CakePHP3アプリ



docker

docker-compose

- [https://github.com/tenkoma/cakephp\\_cms](https://github.com/tenkoma/cakephp_cms)
- Dockerで構築
- CakePHP公式のチュートリアルで作ったCake3アプリにテストを追加

# 開発環境起動

```
$ git clone https://github.com/tenkoma/cakephp_cms.git
```

```
$ cd cakephp_cms
```

```
$ make
```

```
$ make migrate
```

- 設定前にコンテナを起動しておく
- PhpStorm で cakephp\_cms フォルダを開く

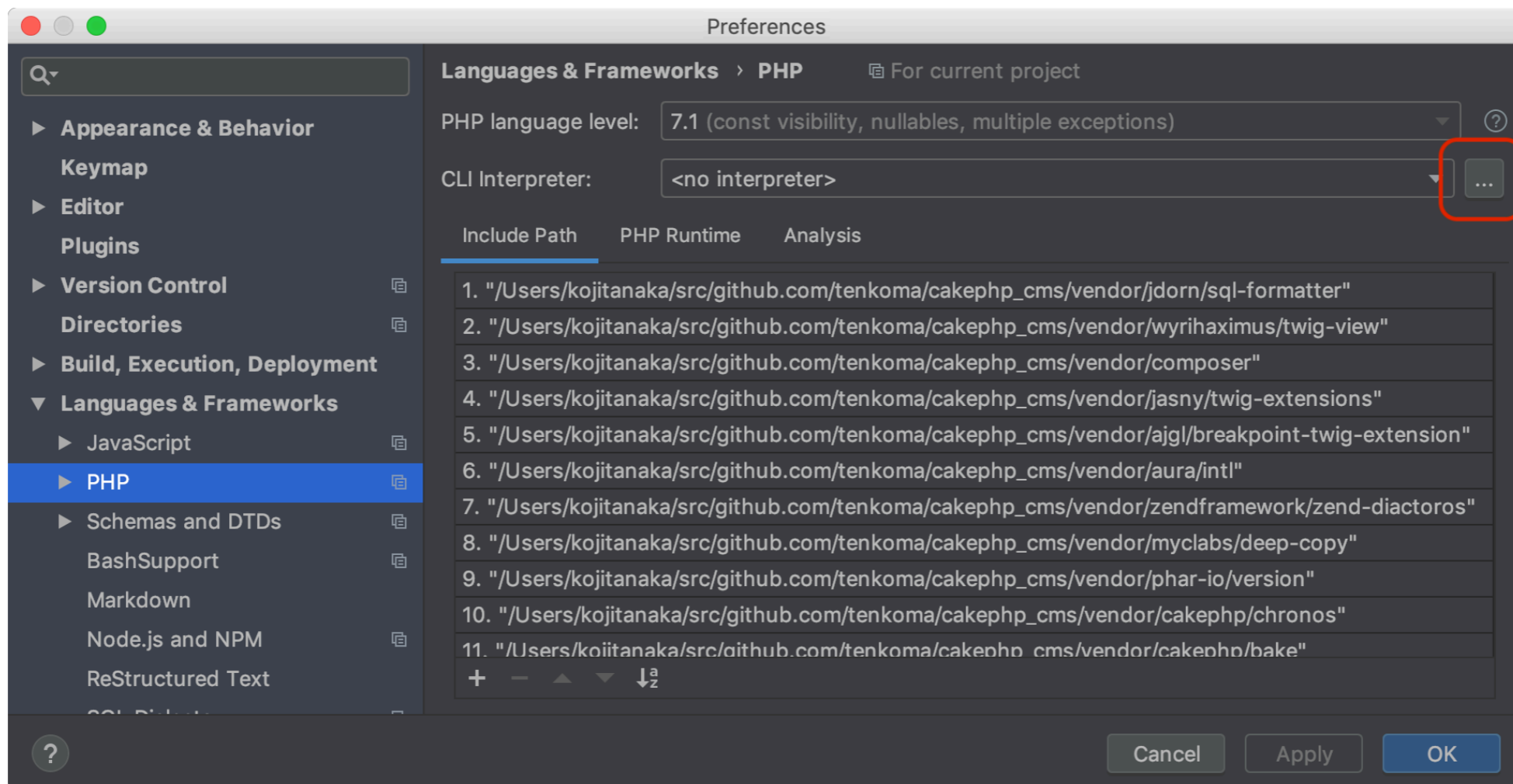
# PHP CLI, Test Frameworks の設定

※かけ足で説明しますが、スクリーンショットを多めに載せているので、後ほどご確認ください



# 設定1 PHP CLI

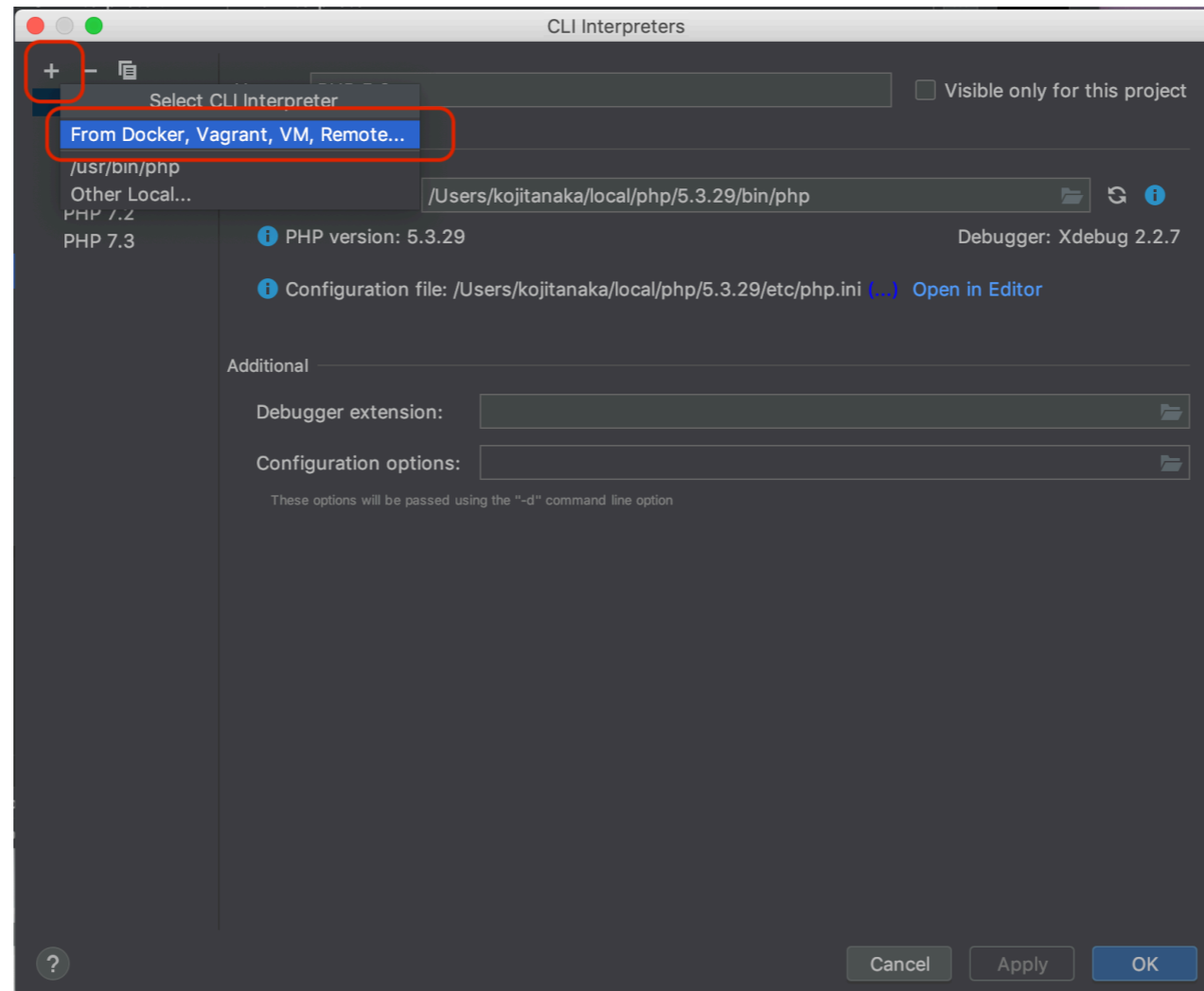
- Preferences > PHP



- CLI Interpreterは未選択になっている
- 「…」を選択

# 設定1 PHP CLI

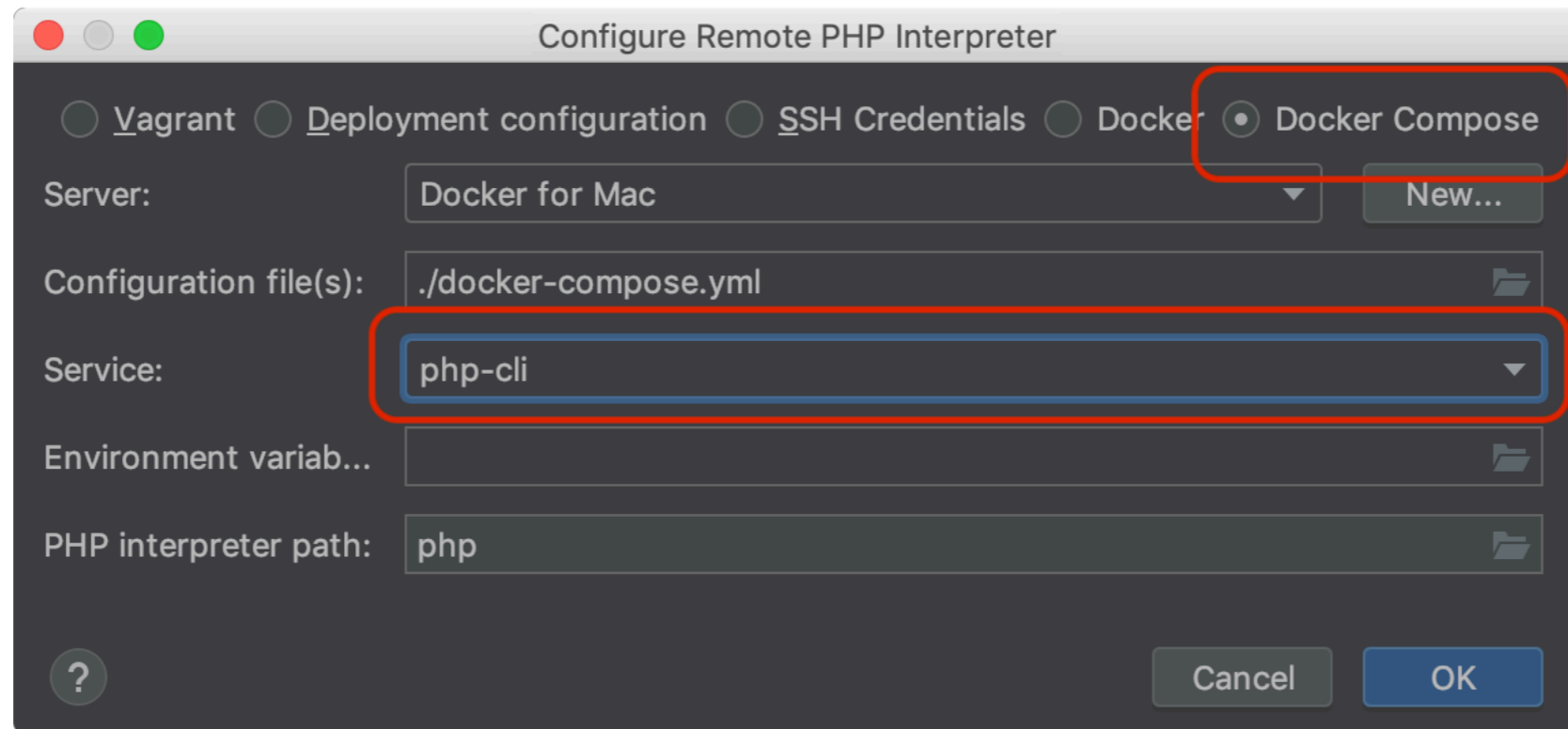
- PHP実行環境を追加する



- 「+」 → 「From Docker…」 を選択

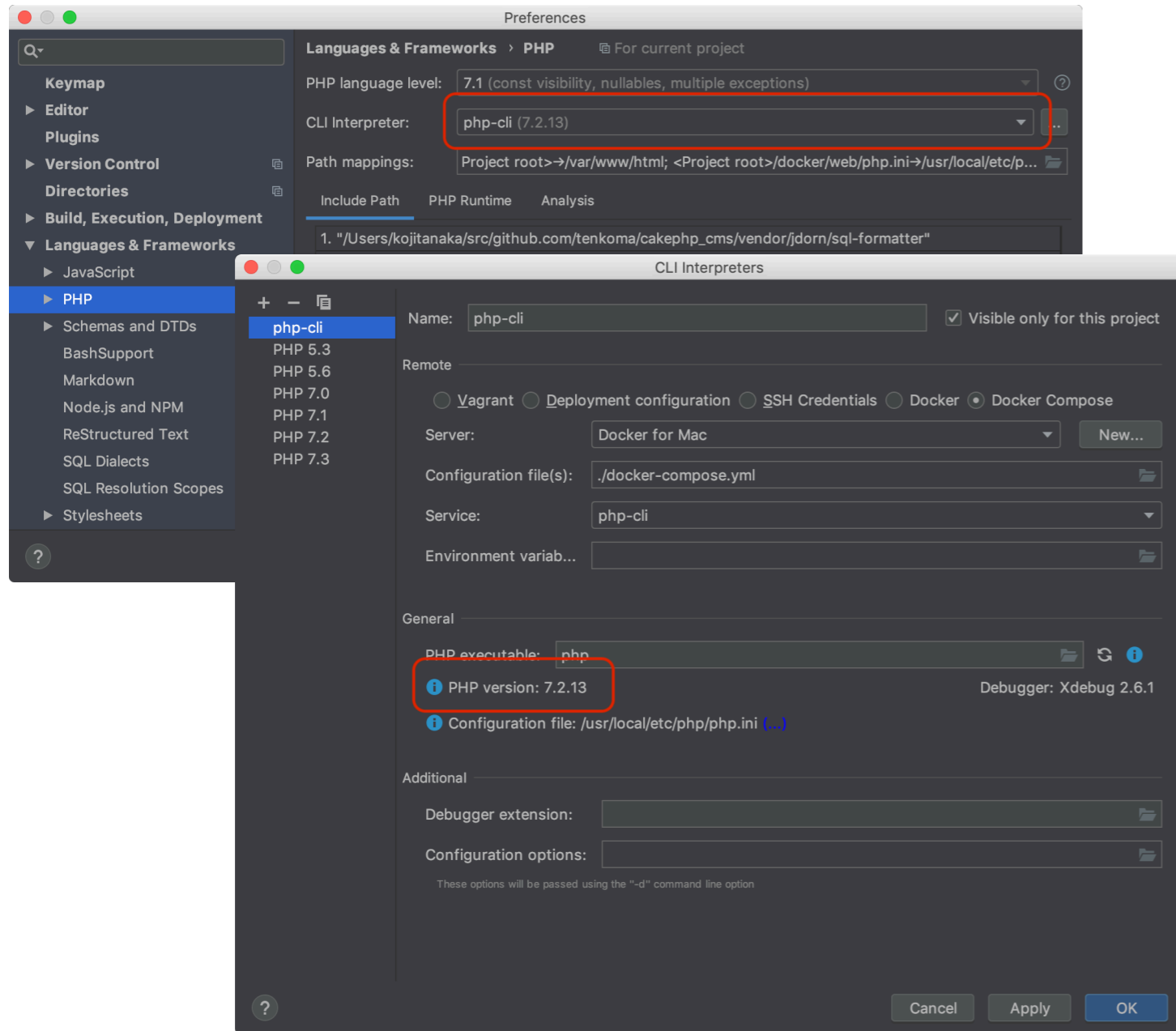
# 設定1 PHP CLI

- リモート実行環境を選択する



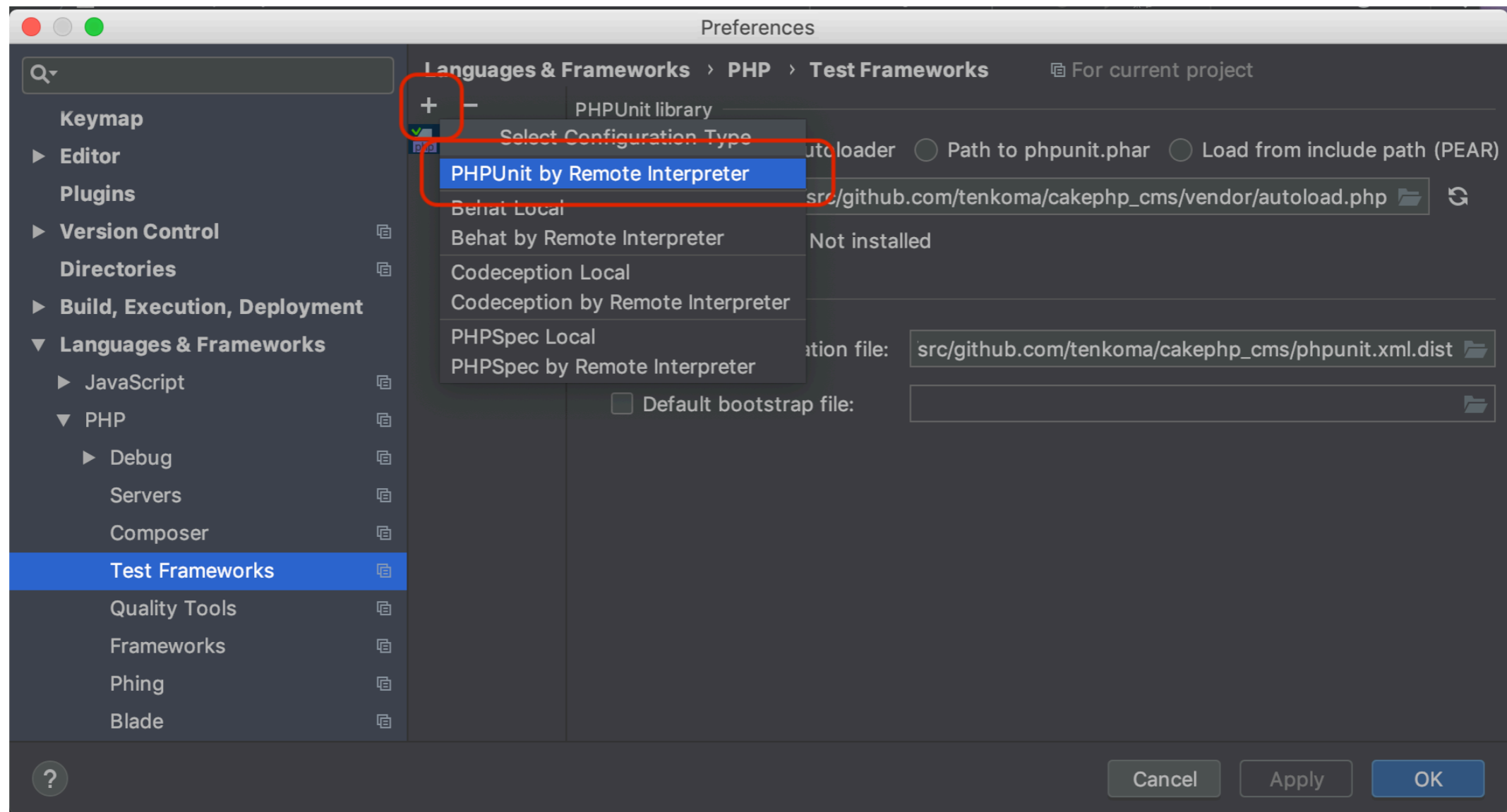
- 「Docker Compose」 → 「php-cli」 を選択して 「OK」

# 設定1 PHP CLI



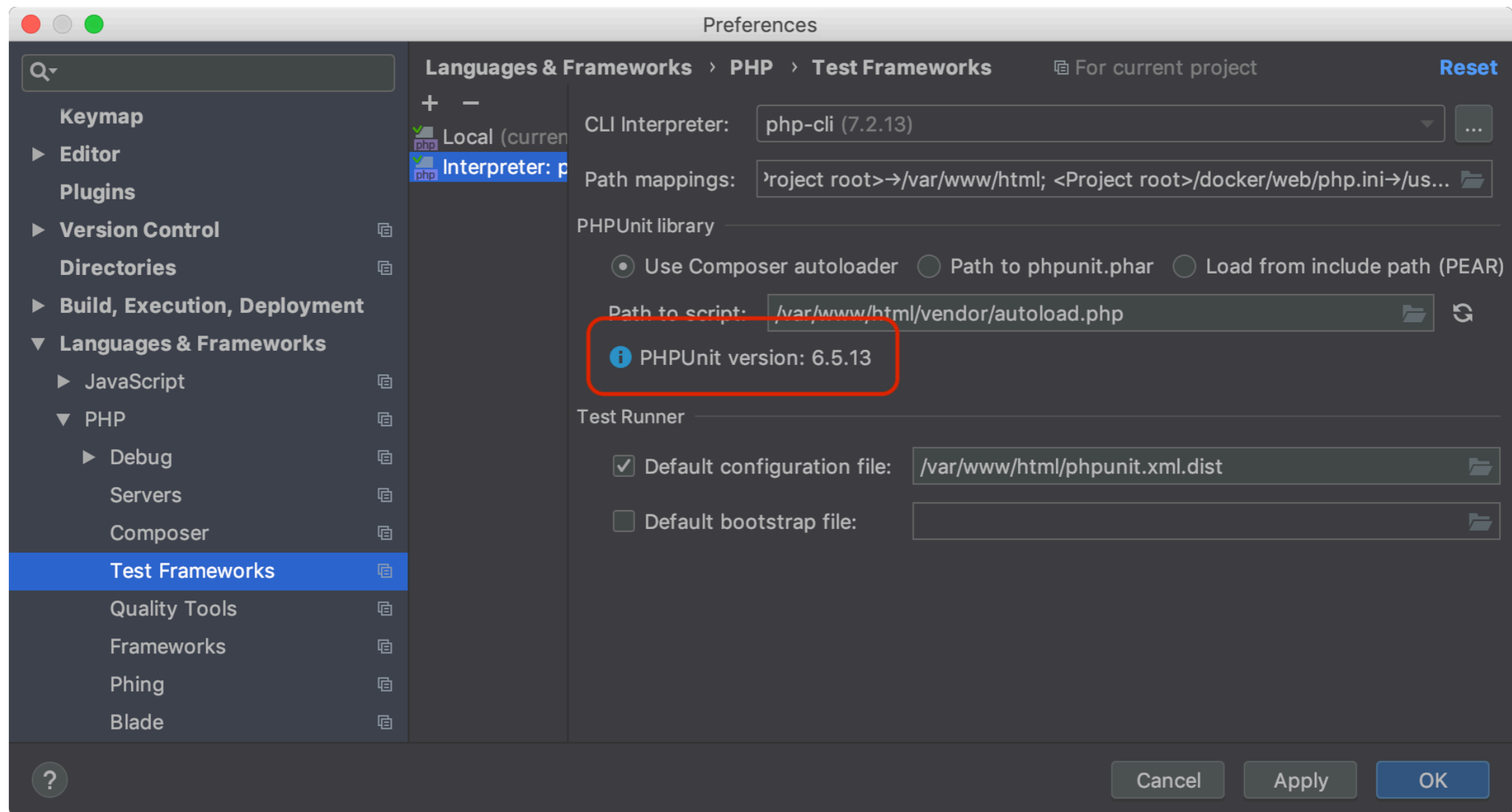
- コンテナ内のPHPバージョンが表示されたらOK
- 表示されなくても大丈夫なこともある

# 設定2 Test Frameworks



- 表示されるダイアログで 'php-cli' を選択してOK
- 'Local' という設定は削除しましょう

# 設定2 Test Frameworks



- PHPUnitのバージョンが表示されたらOK

ユニットテスト作成が  
どう楽になるか

# 1. PhpStormから

## テストを実行する

- テストファイル・テストメソッド・フォルダ内全部
- 失敗したテストのみ再実行
- 失敗したテスト結果からコードジャンプ



デモ

Project

- src
  - tests
    - Fixture
    - TestCase
      - Controller
      - Model
        - Behavior
        - Entity
          - ArticleTest.php
          - UserTest.php
      - Table
      - Routing
      - View

Structure

- UsersControllerTest
  - fixtures: array
  - testLoginShow():void
  - testLoginFailed():void
  - testLoginSucceed():void
  - testLogout():void

```

42         'password' => 'wrongpassword',
43     ]);
44     $this->assertResponseOk();
45     $this->assertResponseContains('content: 'ユーザー名またはパスワードが不正です。');
46 }
47
48 /**
49  * ログイン成功
50  */
51 public function testLoginSucceed()
52 {
53     $this->loginSucceed();
54     $this->assertResponseOk();
55     $this->post(['url' => '/users/login?redirect=%2Farticles%2Fadd', [
56         'email' => 'myname@example.com',
57         'password' => 'password',
58     ]]);
59     $this->assertRedirect(['url' => '/articles/add']);
60     $this->assertSession(['expected' => 1, 'path' => 'Auth.User.id']);
61 }
62
63 /**
64  * ログアウト
65  */

```

Run 'testLoginSucceed' ^⇧F10

Debug 'testLoginSucceed'

Run 'testLoginSucceed' with Coverage

Run: UsersControllerTest

Test Results

App\Test\TestCase\Contr	805 ms
testLoginShow	356 ms
testLoginFailed	261 ms
testLoginSucceed	145 ms
testLogout	43 ms

Tests failed: 1, passed: 3 of 4 tests - 805 ms

```

PHPUNIT 0.5.13 by Sebastian Bergmann and contributors.

Recreating cakephp_cms_php-cli_1 ... done
Failed asserting that response has header 'Location'.
/var/www/html/vendor/cakephp/cakephp/src/TestSuite/IntegrationTestTrait.php
/var/www/html/tests/TestCase/Controller/UsersControllerTest.php:59

```

# メリット

- ターミナルとIDEを切り替えずに済む
- 特定のメソッドだけ実行したいときに `--filter=testSomething` などと指定せずに済む

# 2. テスト実行しながら デバッグ

- 失敗したUsersController::login() (ログイン機能)の途中にブレークポイントを設定して変数を確認してみる
- リクエストでID/PasswordがPostされているか確認(Variablesでインクリメンタルサーチ)
- 「Copy Value As…」で実行中の変数からテストデータを作成

デモ

```
cakephp_cms > tests > TestCase > Controller > UsersControllerTest.php
UsersControllerTest.php x
47
48 /**
49  * ログイン成功
50  */
51 public function testLoginSucceed()
52 {
53     // Run 'testLoginSucceed'
54     // Debug 'testLoginSucceed'
55     // Run 'testLoginSucceed' with Coverage
56     $this->post(['url' => '/users/login?redirect=%2Farticles/add',
57                'email' => 'myname@example.com',
58                'password' => 'password',
59                ]);
60     $this->assertRedirect(['url' => '/articles/add']);
61     $this->assertSession(['expected' => 1, 'path' => 'Auth.User.id']);
62 }
63 /**
64  * ログアウト
65  */
66 public function testLogout()
67 {
68     $this->session(['Auth.User.id' => 1]);
69
70     $this->get('/users/logout');
71
72     $this->assertSession([], 'path' => 'Auth');
73     $this->assertRedirect(['url' => '/users/login']);
74 }
75 }
76

UsersController.php x
107
108 if ($this->Users->delete($user)) {
109     $this->Flash->success(['singular' => 'The user has been deleted.']);
110 } else {
111     $this->Flash->error(['singular' => 'The user could not be deleted.']);
112 }
113
114 return $this->redirect(['action' => 'index']);
115 }
116
117 public function login()
118 {
119     if ($this->request->is(['type' => 'post'])) {
120         $user = $this->Auth->identify();
121         if ($user) {
122             // 認証OK
123             $this->Auth->setUser($user);
124             return $this->redirect($this->Auth->redirectUrl());
125         }
126         // 認証NG
127         $this->Flash->error(['message' => 'ユーザー名またはパスワードが正しくありません。']);
128     }
129 }
130
131 public function logout()
132 {
133     $this->Flash->success(['message' => 'ログアウトしました。']);
134     return $this->redirect($this->Auth->logout());
135 }
136 }
137 }
```

Debug: UsersControllerTest.testLoginSucceed

Debugger | Console

Frames

- UsersController.php:121, App|Controller|UsersController
- Controller.php:610, App|Controller|UsersController
- ActionDispatcher.php:120, Cake\Http\ActionDispatcher
- ActionDispatcher.php:94, Cake\Http\ActionDispatcher
- BaseApplication.php:235, App|Application->invoke()
- Runner.php:65, Cake\Http\Runner->\_\_invoke()
- Runner.php:65, Cake\Http\Runner->\_\_invoke()
- CsrfProtectionMiddleware.php:108, Cake\Http\Middleware

Variables

- request = {Cake\Http\ServerRequest} [23]
- \_detectors = {array} [15]
- params = {array} [10]
- data = {array} [2]
- email = "myname@example.com"
- password = "password"
- query = {array} [1]
- cookies = {array} [1]
- \_environment = {array} [40]
- url = "users/login"

Watches

No watches

# メリット

- ブレークするとテスト中の変数が簡単に確認できる
- 実行中の変数をコピペして簡単にテストデータを作れる

### 3. テスト実行時のカバレッジ をエディタやファイラに表示

- UsersControllerTest で “Run 'tests(PHPUnit)' with Coverage”
- 「Project」 パネルに行カバレッジ%が表示される
- テストできてないコードを見つける
- テストを追加して再実行



デモ

The screenshot displays an IDE interface for a PHP project named 'cakephp\_cms'. The Project panel on the left shows a tree view of the project structure, including folders like 'src', 'Controller', and 'tests'. The main editor shows the code for 'UsersControllerTest.php' and 'UsersController.php'. The test code includes a class 'UsersControllerTest' that extends 'TestCase' and uses 'IntegrationTestTrait'. It has a 'testLoginShow()' method. The code coverage is indicated by a green bar on the left side of the code editor. A red arrow points to line 127 in the test code. The Test Results panel at the bottom shows that all tests passed, with a total of 3 tests and 8 assertions. The test results are as follows:

Test Name	Duration (ms)
testLoginShow	490
testLoginSucceed	342
testLogout	76

- Projectパネル:フォルダやファイルごとの行カバレッジ%が表示される
- 矢印: テストで実行されていない行

# メリット

- コードがテストで実行できているかの調査がしやすく
- どの行が実行できているか確認しながらテスト作成を試行錯誤する
- 行カバレッジ100%を目指さない場合でも便利

# まとめ

PhpStorm とPHPUnit を連携すると

- PhpStorm からテスト実行できて、手間が減る
- デバッグしながらの調査がしやすくなる
- カバレッジを見ながらテストを書くとコードが実行できているか把握しやすい