

PROJECT #1: "Auction Site" v1.0

PART 5: "PHPUnit"

Objectives:

- Install and configure the PHPUnit package
- Use the provided unit test to test the Database class.

Instructions:

In this final part of v1.0, you will install and configure PHPUnit.

PHPUnit is a unit testing framework designed to help you when developing PHP applications by providing you with a easy to use but powerful suite of tools to test your code during development.

Unit testing is an important part of development as it assures you that your code will remain working throughout the development lifecycle.

For the rest of this project, you will be using PHPUnit tests to ensure your code works as expected.

1. Open the **composer.json** file in your project's root directory and add the following code:

```
...  
  
"require": {  
    "monolog/monolog": "^2.0",  
    "ext-PDO": "*",  
    "ext-curl": "*" ,  
},  
"require-dev": {  
    "phpunit/phpunit": "^9.0"  
},  
  
...
```

Note: the phpunit framework is only need during development, therefore can be placed in the **require-dev** section of the composer configuration file.

2. Using the same method in the last part (Part 4), update composer on the server. See Part 4 for more information.
3. Download the provided phpunit configuration file **PHPUnitXml.zip** from blackboard and unpack the file into the project's root directory.
4. Download the provided unit test for the **Database** class; **DatabaseTest.zip** from blackboard and unpack the file into the project's **tests/Unit** directory.

5. Next, drop to the command prompt (Start a new SSH session) and change into the project directory.
6. Before you execute the test, create a shortcut (symbolic link) to the phpunit executable by issuing the following command:

```
[auction]$ ln -s ./vendor/phpunit/phpunit/phpunit phpunit
```

7. You're now ready to execute the unit test. To do so, use the following command:

```
[auction]$ ./phpunit
PHPUnit 9.5.4 by Sebastian Bergmann and contributors.
..... 5 / 5 (100%)

Time: 0:00.072, Memory: 6.00MB

OK (5 tests, 11 assertions)
```

If you encounter an error (`<form action="Undefined index: REQUEST_URI">`), make sure that your `dbfiles_auction/` directory has been placed in the `app/` directory.

8. Review the class files to ensure your project is correctly documented using PHP DocBlocks convention.

Hint: Recall, PhpStorm can generate DocBlocks by pressing **Alt+Enter**

9. Ensure your project follows the correct Git workflow as discussed in class.
-

To summarize what we have done in v1.0:

- Created the project structure and configuration, dependencies and bootstrap files needed.
- Defined the **Database** class including all functionality to create a connection and query the database.
- Created & populated the tables needed for the auction site using the **Database** class we created.
- Implemented the Monolog and PHPUnit frameworks to help with development of the application.

Submit your work for assessment.