

Особенности применения PSR стандартов для оформления корпоративных веб-проектов

Потылицын Андрей Олегович

*Приамурский государственный университет им. Шолом-Алейхема
Студент*

Демьяненко Андрей Анатольевич

*Приамурский государственный университет им. Шолом-Алейхема
Студент*

Глаголев Владимир Александрович

*Приамурский государственный университет имени Шолом-Алейхема
к.г.н., доцент кафедры информационных систем, математики и правовой информатики*

Аннотация

В данной статье рассматривается анализ программного обеспечения, позволяющего проверять правильность написания PHP скриптов на стандарт PSR с помощью программы JetBrains PhpStorm.

Ключевые слова: PHP, PSR, код, валидация, тестирование

Features of the application of PSR standards for the design of corporate web projects

Potylitsyn Andrey Olegovich

*Sholom-Aleichem Priamursky State University
Student*

Demyanenko Andrey Anatolyevich

*Sholom-Aleichem Priamursky State University
Student*

Glagolev Vladimir Aleksandrovich

*Sholom-Aleichem Priamursky State University
candidate of geographical Sciences, associate Professor of the Department of Information Systems, Mathematics and Legal Informatics*

Abstract

This article discusses software analysis that allows you to verify the correctness of writing PHP scripts to the PSR standard using the JetBrains PhpStorm program.

Keywords: PHP, PSR, code, validation, testing.

Для проверки правильности программного кода и разработки корпоративного программного обеспечения требуется специальным образом оформленный проект, для дальнейшего его развития и возможность подключения сторонних программистов необходимо применение современных средств документирования кода.

Целью работы является анализ существующих средств программного кода, написанного на языке PHP и проверка тестового дипломного проекта на его соответствие международным стандартам.

В настоящее время существует пять стандартов PHP кода: PSR-0, PSR-1, PSR-2, PSR-3, PSR-4 [1,2].

Так стандарт автозагрузки PSR-0 обязателен к исполнению в целях обеспечения совместимости механизмов автозагрузки.. Данный стандарт полностью определяет структуру документа, которая должна выглядеть следующим образом: `\<Vendor Name>\(<Namespace>)*<Class Name>`. При обращении к файловой системе полностью определённое пространство имён и имя класса дополняются суффиксом `.php`.

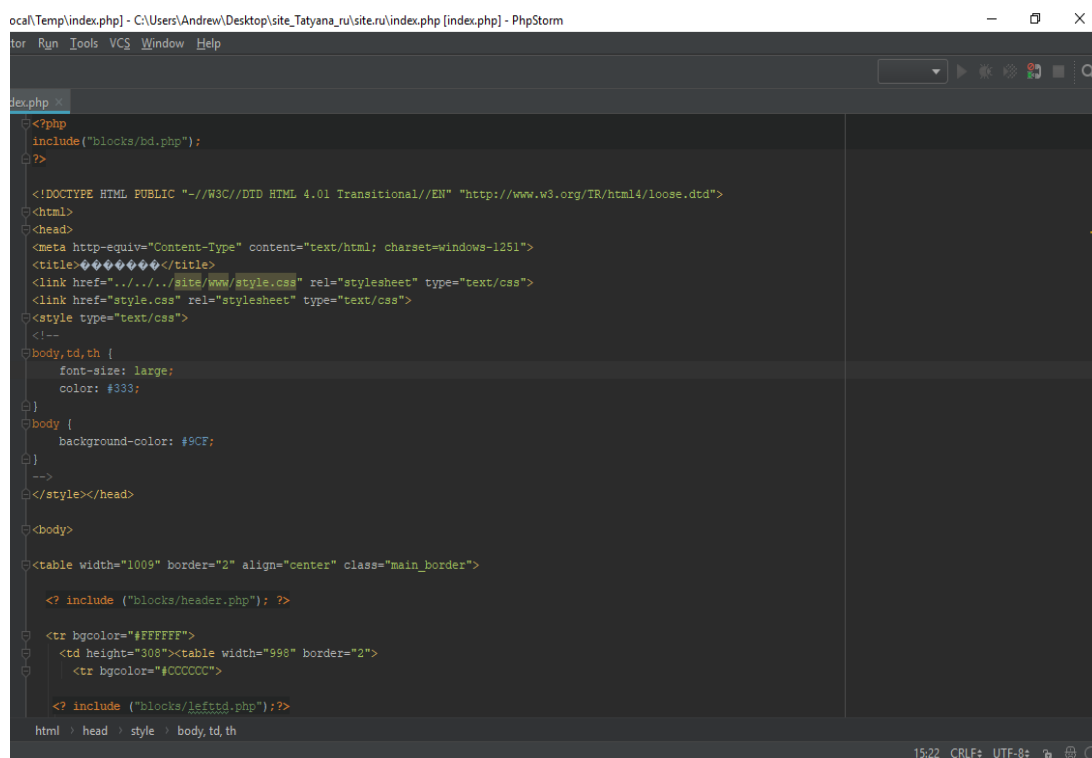
Базовый стандарт PSR-1 содержит исполняемый код, описываются стандартные элементы, которые необходимы для обеспечения высокой технической совместимости кода, созданного, либо поддерживаемого уровня разработчиков.

Для восприятия кода различными разработчиками помимо стандарта PSR-1 используется PSR-2 на основе управляющих кронштейнов. Интерфейс библиотек протоколирования задается стандартом PSR-3. В случае, если определенная инфраструктура или CMS нуждается в расширенной функциональности при разработке приложения подключают централизованную систему ведения журнала.

Стандарт отвечает PSR-4 за расширенное автоматическое резервное копирование, т.е. описывает спецификацию классов запуска на основе путей к файлам проекта. Он полностью совместим с любой другой спецификацией запуска, включая PSR-0. В этом стандарте также описываются правила размещения файлов для автозагрузки.

В ходе работы был проанализирован дипломный проект на соответствие стандартам оформления PHP кода. Для анализа мы использовали коммерческую кроссплатформенную интегрированную среду разработки для PHP компании JetBrains на основе платформы IntelliJ IDEA (рис. 1).

В дипломном проекте содержится 23 файла с расширением `.php`. Размер дипломного проекта составляет 159 Кбайт.



```
<?php
include("blocks/bd.php");
?>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1251">
<title><<<<<<<<<<<<</title>
<link href="../../site/www/style.css" rel="stylesheet" type="text/css">
<link href="style.css" rel="stylesheet" type="text/css">
<style type="text/css">
<!--
body,td,th {
font-size: large;
color: #333;
}
body {
background-color: #9CF;
}
-->
</style></head>
<body>
<table width="100%" border="2" align="center" class="main_border">
<? include ("blocks/header.php"); ?>
<tr bgcolor="#FFFFFF">
<td height="308"><table width="998" border="2">
<tr bgcolor="#CCCCCC">
<? include ("blocks/lefttd.php");?>

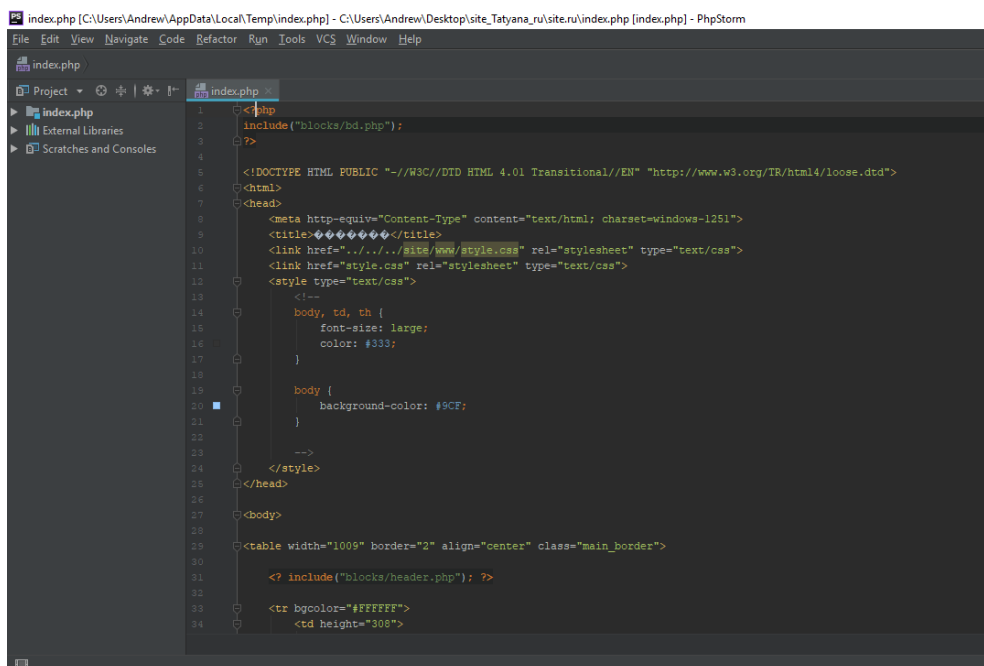
```

Рис.1 Внешний вид программы PhpStorm

Для оформления данного кода в PhpStorm нажимаем на кнопку File. В раскрывшемся меню находим пункт Settings. Нажав на Settings, в появившемся окне находим команду Editor. Нажав на него, раскрывается список, в котором находим пункт Code Style. Затем нажимаем на пункт Code Style и выбираем язык PHP. Далее находим пункт PHPDoc и ставим галочку в команде Blank line before the first tag. Необходимо оформить код в форматах PSR-1/PSR-2. Нажимаем Set from..., далее на кнопку Predefined Style и затем нажимаем на пункт PSR1/PSR2. Затем нажимаем на кнопку Apply и Ok.

После настройки оформления кода нам нужно проверить на соответствие данный код на соответствие форматам PSR-1 и PSR-2. В строке меню находим пункт Code и используем команду Reformat Code.

После применения данной команды код стал таким, как показано на рисунке ниже (рис. 2).



```
index.php [C:\Users\Andrew\AppData\Local\Temp\index.php] - C:\Users\Andrew\Desktop\site_Tatyana_ru\site_ru\index.php [index.php] - PhpStorm
File Edit View Navigate Code Refactor Run Tools VCS Window Help
index.php
Project
index.php
External Libraries
Scratches and Consoles
1 <?php
2 include("blocks/bd.php");
3 ?>
4
5 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
6 <html>
7 <head>
8 <meta http-equiv="Content-Type" content="text/html; charset=windows-1251">
9 <title>?????????</title>
10 <link href="../../site/www/style.css" rel="stylesheet" type="text/css">
11 <link href="style.cas" rel="stylesheet" type="text/css">
12 <style type="text/css">
13 <!--
14     body, td, th {
15         font-size: large;
16         color: #333;
17     }
18
19     body {
20         background-color: #9CF;
21     }
22 -->
23 </style>
24 </head>
25 </html>
26
27 <body>
28
29 <table width="100%" border="2" align="center" class="main_border">
30
31 <? include("blocks/header.php"); ?>
32
33 <tr bgcolor="#FFFFFF">
34 <td height="308">
```

Рис. 2. Результат оформления

Таким образом, среда программирования PHPStorm позволяет проверить программный код на ошибки, провести рефакторинг сайта, осуществлять поддержку и редактирования шаблонов smarty (подсвечивание синтаксических ошибок, автодополнение функций и атрибутов Smarty, автоматическая вставка парных скобок, кавычек и закрывающих тегов и др.), реализовывать MVC представление для Symfony2 и Yii фреймворков.

Библиографический список

1. Стандарты PSR https://svyatoslav.biz/misc/psr_translation/ (дата обращения 16.06.2018)
2. PHPStorm <https://www.jetbrains.com/phpstorm/> (дата обращения 16.06.2018)