

Министерство науки и высшего образования Российской Федерации
Севастопольский государственный университет
Кафедра ИС

Отчет
по лабораторной работе №8
«Исследование архитектуры MVC приложения и возможностей обработки
данных HTML-форм на стороне сервера с использованием языка PHP»
по дисциплине
«ВЕБ-ТЕХНОЛОГИИ»

Выполнил студент группы ИС/б-17-2-о
Горбенко К. Н.
Проверил
Овчинников А.Л.

Севастополь
2020

1 ЦЕЛЬ РАБОТЫ

Изучить основы синтаксиса PHP и принципы функционирования MVC приложения на стороне сервера. Приобрести практические навыки использования языка PHP для генерации HTML-кода и обработки HTML-форм в MVC приложении.

2 ЗАДАНИЕ НА РАБОТУ

1. В соответствии с рекомендациями, представленными в п. 1.2.8 реструктурировать код пресонального сайта на основе шаблона MVC.

2. Для страницы «Фотоальбом» реализовать хранение данных фото (подписей и имен файлов) в соответствующих константах модели Photo. Вывод таблицы, содержащей фото, реализовать в представлении с использованием операторов циклов.

3. Реализовать класс FormValidation, выполняющий валидацию данных форм, передаваемых на сторону сервера.

4. С использованием разработанного класса реализовать валидацию форм `enquoteКонтакт` и `enquoteТест` по дисциплине.

5. Реализовать дочерний класс CustomFormValidation от класса FormValidation, дополнив его возможностью выполнения специализированной проверки формы «Тест по дисциплине» на стороне сервера.

6. Реализовать дочерний класс ResultsVerification от класса CustomFormValidation, дополнив его возможностью проверки правильности ответов, введенных пользователем на странице «Тест по дисциплине» (реализовать проверку правильности для вопросов с элементами ввода типа RadioButton, ComboBox или однострочный текст) и вывода результатов проверки пользователю.

3 ХОД РАБОТЫ

3.1 Реализация маршрутизации

Для реализации MVC архитектуры создадим маршрутизацию вида `site/controller/action`, для этого используем `UrlRewrite` модуль и создадим входную точку приложения `index.php`.

`.htaccess`:

```
1 RewriteEngine On
2 RewriteCond %{REQUEST_FILENAME} !-f
3 RewriteCond %{REQUEST_FILENAME} !-d
4 RewriteRule ^(.*)/(.*) index.php?controller=$1&action=$2 [QSA]
```

`index.php`:

```
1 <?php
2 include "Core/Router.php";
3
4 $request = $_REQUEST;
5 $requestMethod = $_SERVER["REQUEST_METHOD"];
6 Router::route($request, $requestMethod);
```

Теперь реализуем роутер, с разной маршрутизацией для GET и POST запросов.

`Core/Router.php`

```
1 class Router
2 {
3     protected static $actionPrefixesByMethod = array(
4         "GET" => "",
5         "POST" => "Post"
6     );
7
8     static function route($request, $requestMethod) {
9         $controllerName = self::getControllerName($request);
10        $actionName = self::getActionName($request, $requestMethod);
11
12        if ($requestMethod != "GET" && $requestMethod != "POST") {
13            http_response_code(ResponseCodes::$notAllowedMethodStatusCode);
14            die();
15        }
16
17        $controllerFileName = "Controllers/" . $controllerName . 'Controller.
18            php';
19        if (file_exists($controllerFileName)) {
20            include $controllerFileName;
21        } else {
```

```

21         http_response_code(ResponseCodes::$NotFoundStatusCode);
22         die();
23     }
24
25     $controllerClassName = ucfirst($controllerName) . 'Controller';
26     $container = Container::getInstance();
27     $controller = new $controllerClassName($container);
28
29     if (method_exists($controller, $actionName)) {
30         $controller->$actionName();
31     } else {
32         http_response_code(ResponseCodes::$NotFoundStatusCode);
33         die();
34     }
35 }
36
37 private static function getControllerName($request) {
38     return array_key_exists("controller", $request)
39         ? $request["controller"]
40         : "Home";
41 }
42
43 private static function getActionName($request, $method) {
44     if (!array_key_exists("action", $request)) {
45         return "Index" . self::$actionPrefixesByMethod[$method];
46     }
47
48     return $request["action"] . self::$actionPrefixesByMethod[$method];
49 }
50 }

```

Данный класс, учитывая имена контроллера и экшена из строки параметров, создает экземпляр соответствующего контроллера и вызывает соответствующий экшен, если они существуют. Иначе, он возвращает 404 ответ.

3.2 Реализация MVC

Архитектура MVC состоит из контроллеров, моделей и представлений. Создадим первый контроллер:

Controllers/HomeController.php:

```

1 class HomeController
2 {
3     private $photosRepository;
4
5     public function __construct(IContainer $container) {

```

```

6         $this->photosRepository = $container->resolve("IPhotosRepository");
7     }
8
9     public function index() {
10         ViewRenderer::render("Views/Home/Index.php", "Homepage");
11     }
12
13     public function about() {
14         $viewModel = new AboutViewModel($this->photosRepository->getAll());
15         ViewRenderer::render("Views/Home/About.php", "About me", $viewModel);
16     }
17
18     public function interests() {
19         ViewRenderer::render("Views/Home/Interests.php", "My interests");
20     }
21
22     public function learning() {
23         ViewRenderer::render("Views/Home/Learning.php", "Learning");
24     }
25
26     public function photos() {
27         ViewRenderer::render("Views/Home/Photos.php", "Photo album");
28     }
29
30     public function history() {
31         ViewRenderer::render("Views/Home/History.php", "History");
32     }
33 }

```

ViewRenderer.php:

```

1 class ViewRenderer
2 {
3     static function render($viewName, $title, $viewModel = NULL, $layout = '
        Layout.php')
4     {
5         include "Views/Shared/".$layout;
6     }
7 }

```

Views/Shared/Layout.php:

```

1 <!DOCTYPE html>
2
3 <html lang="en">
4     <head>
5         <title><?=$title?></title>
6         <meta content="width=device-width" name="viewport" />
7         <meta content="no-cache" http-equiv="Cache-Control" />

```

```

8      <link href="../../../client-side/bundles/layout.bundle.css" rel="stylesheet"
      type="text/css" />
9  </head>
10 <body>
11     <nav>
12         <ul>
13             <li><span>Web</span></li>
14             <li><a href="/Home/Index">Home</a></li>
15             <li><a href="/Home/About">About</a></li>
16             <li><a href="/Home/Interests">My interests</a></li>
17             <li><a href="/Home/Learning">Learning</a></li>
18             <li><a href="/Home/Photos">Photos</a></li>
19             <li><a href="/Contact/Index">Contact</a></li>
20             <li><a href="/Test/Index">Test</a></li>
21             <li><a href="/Home/History">History</a></li>
22             <li><span id="time"></span><br/><span id="date"></span></li>
23         </ul>
24     </nav>
25     
26
27     <div class="main-content">
28         <?php include($viewName); ?>
29     </div>
30
31     <footer class="full-width">
32         <section>
33             <header><h2>Footer</h2></header>
34             <p>Some useful links here</p>
35             <ul>
36                 <li>Link</li>
37                 <li>Another link</li>
38             </ul>
39             <p>
40                 And some info as well: Nunc eget viverra neque, non
41                 ullamcorper odio. Integer dolor sem, ultrices eu
42                 tempus dapibus, sagittis ac lacus. Duis rutrum turpis eu
43                 lectus imperdiet, ac
44                 sagittis orci faucibus. Vestibulum volutpat fringilla
45                 turpis vehicula lobortis.
46                 Interdum et malesuada fames ac ante ipsum primis in
47                 faucibus. Aliquam posuere
48                 tempor enim, et tempus tellus commodo mattis. Sed
49                 sollicitudin mauris vitae
50                 vestibulum malesuada.
51             </p>
52         </section>
53     </footer>

```

```

49         <script src="../../client-side/bundles/layout.bundle.js" >
50     </body>
51 </html>

```

Views/Home/Index.php

```

1 <link href="../../client-side/bundles/home/index.bundle.css" rel="stylesheet" type
  ="text/css" />
2
3 <header><h1>Portfolio</h1></header>
4 <section>
5     <header><h2>About me</h2></header>
6     <p><i>Gorbenko Kirill Nikolaevich.</i></p>
7 </section>
8 <section>
9     <header><h2>NEXT CHAPTER</h2></header>
10 </section>
11
12 <script src="../../client-side/bundles/home/index.bundle.js"></script>

```

Views/Home/About.php:

```

1 <link href="../../client-side/bundles/home/about.bundle.css" rel="stylesheet" type
  ="text/css" />
2
3 <header>
4     <h1>Autobiography</h1>
5 </header>
6 <section>
7     <header>
8         <h1>Section 1</h1>
9     </header>
10    <p>There should be some authobiography</p>
11    <ol>
12        <li>19 years old</li>
13        <li>My parents's son.</li>
14        <li>Dunno</li>
15    </ol>
16    <p>Photos:</p>
17    <div class="photo-wrapper full-width">
18        <?php
19            foreach ($viewModel->photos as $photoName => $path) {
20                echo("<img src='$path' alt='$photoName' />");
21            }
22        ?>
23    </div>
24 </section>
25
26 <script src="../../client-side/bundles/home/about.bundle.js"></script>

```

Views/Home/Interests.php:

```

1 <link href="../../../client-side/bundles/home/interests.bundle.css" rel="stylesheet"
  type="text/css" />
2
3 <header>
4   <h1>My interests</h1>
5 </header>
6 <section>
7   <header>
8     <h2>Contents</h2>
9   </header>
10  <button class="modal-button" id="dialog">Modal</button>
11  <ul id="interests">
12    <li class="contents" id="Hobbies">My hobbies</li>
13    <li class="interests" id="Hobbies-expand">
14      <h2>My hobbies</h2>
15      <p>My hobbies are supposed to be here</p>
16    </li>
17    <li class="contents" id="Books">My books</li>
18    <li class="interests" id="Books-expand">
19      <h2>My books</h2>
20      <p>My books are supposed to be here</p>
21    </li>
22    <li class="contents" id="Music">My music</li>
23    <li class="interests" id="Music-expand">
24      <h2>My music</h2>
25      <p>My music is supposed to be here</p>
26    </li>
27    <li class="contents" id="Films">My films</li>
28    <li class="interests" id="Films-expand">
29      <h2>My films</h2>
30      <p>My films are supposed to be here</p>
31    </li>
32  </ul>
33 </section>
34
35 <div class="modal">
36   <div class="modal-window">
37     <div>
38       <p class="modal-message"></p>
39     </div>
40     <div class="modal-navigation">
41       <button class="modal-button modal-confirm-button"></button>
42       <button class="modal-button modal-cancel-button"></button>
43     </div>
44   </div>
45 </div>
46

```



```
47 <script src="../../client-side/bundles/home/interests.bundle.js"></script>
```

Views/Home/Learning.php:

```
1 <link href="../../client-side/bundles/home/learning.bundle.css" rel="stylesheet"
  type="text/css" />
2
3 <section>
4   <header>
5     <h1>Disciplines</h1>
6   </header>
7   <h2>Институт информационных технологий и управления в технических системах
    </h2>
8   <h2>Информационные системы и технологии</h2>
9   <table>
10    <tr>
11      <td colspan="9">Disciplines by semesters</td>
12    </tr>
13    <tr>
14      <th scope="row">First</th>
15      <td>Math</td>
16      <td>AIP</td>
17      <td>Info</td>
18      <td>Hitory</td>
19      <td>Physycs</td>
20      <td>Ecology</td>
21      <td>Descrete Math</td>
22      <td>English</td>
23    </tr>
24    <tr>
25      <th scope="row">Second</th>
26      <td>Math</td>
27      <td>AIP</td>
28      <td>Physics</td>
29      <td>Philosophy</td>
30      <td>Rights</td>
31      <td>Economics</td>
32      <td>BZD</td>
33      <td>English</td>
34    </tr>
35    <tr>
36      <th scope="row">Third</th>
37      <td>Architecture</td>
38      <td>Math</td>
39      <td>Discrete math</td>
40      <td>OOP</td>
41      <td>TEC</td>
42      <td>Economics</td>
43      <td>Graphics</td>
```

```

44         <td>English</td>
45     </tr>
46     <tr>
47         <th scope="row">Fourth</th>
48         <td>Electronics</td>
49         <td>Algorithms</td>
50         <td>Probability</td>
51         <td>English</td>
52         <td>System analysis</td>
53         <td>Databases</td>
54         <td>TSPP</td>
55         <td>OS</td>
56     </tr>
57 </table>
58 </section>
59
60 <script src="../../client-side/bundles/home/learning.bundle.js"></script>

```

Views/Home/Photos.php:

```

1 <link href="../../client-side/bundles/home/photos.bundle.css" rel="stylesheet"
  type="text/css" />
2
3 <header>
4     <h1>Photoalbum</h1>
5 </header>
6 <div class="photo-wrapper full-width"></div>
7
8 <div id="lightbox">
9     <div class="lightbox-content">
10         <div class="image-wrapper">
11             <img src="" />
12         </div>
13     <div>
14         <div class="lightbox-button lightbox-arrow-left">&lt;</div>
15         <div class="lightbox-button lightbox-arrow-right">&gt;>/div>
16     </div>
17 </div>
18 <div class="lightbox-close×"></div>
19 </div>
20
21 <script src="../../client-side/bundles/home/photos.bundle.js"></script>

```

Views/Home/History.php:

```

1 <link href="../../client-side/bundles/home/history.bundle.css" rel="stylesheet"
  type="text/css" />
2
3 <header>

```

```

4      <h1>History</h1>
5 </header>
6 <section>
7     <header>
8         <h2>Session history</h2>
9     </header>
10    <table class="session-history">
11        <tr>
12            <th colspan="2">Session history</th>
13        </tr>
14        <tr>
15            <td>Home page</td>
16            <td class="home-page"></td>
17        </tr>
18        <tr>
19            <td>About me</td>
20            <td class="about-page"></td>
21        </tr>
22        <tr>
23            <td>My interests</td>
24            <td class="interests-page"></td>
25        </tr>
26        <tr>
27            <td>Learning</td>
28            <td class="learning-page"></td>
29        </tr>
30        <tr>
31            <td>Photo</td>
32            <td class="photos-page"></td>
33        </tr>
34        <tr>
35            <td>Contact</td>
36            <td class="contact-page"></td>
37        </tr>
38        <tr>
39            <td>Test</td>
40            <td class="test-page"></td>
41        </tr>
42        <tr>
43            <td>History</td>
44            <td class="history-page"></td>
45        </tr>
46    </table>
47 </section>
48 <section>
49     <header>
50         <h2>Day History</h2>
51     </header>

```

```

52     <table class="global-history">
53         <tr>
54             <th colspan="2">All-time history</th>
55         </tr>
56         <tr>
57             <td>Home page</td>
58             <td class="home-page"></td>
59         </tr>
60         <tr>
61             <td>About me</td>
62             <td class="about-page"></td>
63         </tr>
64         <tr>
65             <td>My interests</td>
66             <td class="interests-page"></td>
67         </tr>
68         <tr>
69             <td>Learning</td>
70             <td class="learning-page"></td>
71         </tr>
72         <tr>
73             <td>Photo</td>
74             <td class="photos-page"></td>
75         </tr>
76         <tr>
77             <td>Contact</td>
78             <td class="contact-page"></td>
79         </tr>
80         <tr>
81             <td>Test</td>
82             <td class="test-page"></td>
83         </tr>
84         <tr>
85             <td>History</td>
86             <td class="history-page"></td>
87         </tr>
88     </table>
89 </section>
90
91 <script src="../../client-side/bundles/home/history.bundle.js"></script>

```

3.3 Реализация валидации форм

Создадим контроллер с экшеном, отрисовывающим форму.

Controllers/ContactController.php:

```

1 class ContactController

```

```

2 {
3     public function __construct(IContainer $container) {}
4
5     public function index() {
6         $viewModel = new ValidationViewModel(true, array());
7         ViewRenderer::render("Views/Contact/Index.php", "Contact me",
8             $viewModel);
9     }
10
11     public function contactPost() {
12         $validator = new ContactRequestValidator($_POST);
13         $validationResult = $validator->validate();
14
15         if ($validationResult->isValid) {
16             ViewRenderer::render("Views/Contact/Success.php", "Contact me");
17             return;
18         }
19
20         $viewModel = new ValidationViewModel($validationResult->isValid,
21             $validationResult->errors);
22         ViewRenderer::render("Views/Contact/Index.php", "Contact me",
23             $viewModel);
24     }
25 }

```

Views/Contact/Index.php:

```

1 <link href="../../../client-side/bundles/contact/index.bundle.css" rel="stylesheet"
2     type="text/css" />
3 <section>
4     <header>
5         <h1>Mail Me!</h1>
6     </header>
7     <p>All fields are required! Hover over fields to see the additional
8         requirements.</p>
9
10     <?php ViewHelper::writeValidationErrorsList($viewModel->errors); ?>
11
12     <form action="Contact" method="POST">
13         <ol>
14             <li>
15                 <label for="full-name-input">Full Name</label>
16                 <input class="field-long" id="full-name-input" name="Name" type=
17                     ="text" />
18                 <br/>
19             </li>
20             <li>
21                 <label for="email-input">Email</label>

```

```

20         <input class="field-long" id="email-input" name="Email" type="
           email" />
21     <br/>
22 </li>
23 <li>
24     <label for="phone-input">Phone number</label>
25     <input class="field-long" id="phone-input" name="Phone" type="
           text" />
26     <br/>
27 </li>
28 <li>
29     <label for="message-input">Your Message</label>
30     <textarea class="field-long" id="message-input" name="Message
           "></textarea>
31 </li>
32 <li>
33     <label for="datepicker-input">Date</label>
34     <input class="field-long" id="datepicker-input" name="Date"
           type="text" />
35     <br/>
36 </li>
37 <li>
38     <input type="submit" value="Submit" />
39     <br/>
40 </li>
41 </ol>
42 </form>
43 </section>
44
45 <script src="../../client-side/bundles/contact/index.bundle.js"></script>

```

Views/Contact/Success.php

```

1 <link href="../../client-side/bundles/contact/success.bundle.css" rel="stylesheet"
  type="text/css" />
2
3 <h1>Thank you for your appeal!</h1>
4
5 <h2>I will try to contact you back as soon as I can!</h2>
6
7 <script src="../../client-side/bundles/contact/success.bundle.js"></script>

```

Реализуем валидацию объекта `$_POST` с помощью библиотеки Valitron.

Validators/BaseRequestValidator.php:

```

1 abstract class BaseRequestValidator {
2     protected array $post;
3

```

```

4     public function __construct($post) {
5         if (empty($post)) {
6             http_response_code(ResponseCodes::$badRequestStatusCode);
7             die();
8         }
9         $this->post = $post;
10    }
11 }

```

Validators/ContactRequestValidator.php:

```

1 class ContactRequestValidator extends BaseRequestValidator {
2     public function __construct($post) {
3         parent::__construct($post);
4     }
5
6     public function validate() {
7         $v = new Validator($this->post);
8         $v->rule('required', ['Name', 'Email', 'Phone', 'Message', 'Date']);
9         $v->rule('regex', 'Name', '/[A-Za-z]+ [A-Za-z]+ [A-Za-z]+/');
10        $v->rule('email', 'Email');
11        $v->rule('regex', 'Phone', '/^[+][7|3]\d{9,11}$/' );
12        $v->rule('dateFormat', 'Date', 'd.m.Y');
13
14        $isValid = $v->validate();
15        $errors = $v->errors();
16
17        return new ValidationResult($isValid, $errors);
18    }
19 }

```

Validators/ValidationResult.php:

```

1 class ValidationResult {
2     public bool $isValid;
3     public array $errors;
4
5     public function __construct(bool $isValid, array $errors) {
6         $this->isValid = $isValid;
7         $this->errors = $errors;
8     }
9 }

```

Создадим контроллер, обрабатывающий тест:

Controllers/TestController.php:

```

1 class TestController {
2     public function __construct(IContainer $container) {}
3

```

```

4     public function index() {
5         $viewModel = new ValidationViewModel(true, array());
6         ViewRenderer::render("Views/Test/Index.php", "Test", $viewModel);
7     }
8
9     public function testPost() {
10        $validator = new TestRequestValidator($_POST);
11        $validationResult = $validator->validate();
12
13        if ($validationResult->isValid) {
14            $testVerification = new TestRequestVerification($_POST);
15            $viewModel = new SuccessfulTestViewModel($testVerification->verify
                ());
16            ViewRenderer::render("Views/Test/Success.php", "Test", $viewModel);
17            return;
18        }
19
20        $viewModel = new ValidationViewModel($validationResult->isValid,
            $validationResult->errors);
21        ViewRenderer::render("Views/Test/Index.php", "Test", $viewModel);
22    }
23 }

```

Validators/TestRequestValidator.php:

```

1 class TestRequestValidator extends BaseRequestValidator {
2     public function __construct($post) {
3         parent::__construct($post);
4     }
5
6     public function validate() {
7         $v = new Validator($this->post);
8         $v->rule('required', ['Question1', 'Question2', 'Question3', 'Name', '
            Email' ]);
9         $v->rule('integer', 'Question1')->rule('in', 'Question1', [ 1, 2 ]);
10        $v->rule('integer', 'Question2')->rule('in', 'Question2', [ 1, 2, 3 ]);
11        $v->rule('regex', 'Name', '/[A-Za-z]+ [A-Za-z]+ [A-Za-z]+/');
12        $v->rule('email', 'Email');
13
14        $isValid = $v->validate();
15        $errors = $v->errors();
16
17        return new ValidationResult($isValid, $errors);
18    }
19 }

```

Validators/TestRequestVerification.php:

```

1 class TestRequestVerification {

```



```

2     private array $post;
3
4     public function __construct($post) {
5         if (empty($post)) {
6             throw new InvalidArgumentException("$post");
7         }
8         $this->post = $post;
9     }
10
11    public function verify() {
12        $correctAnswers = array(
13            'Question1' => '1',
14            'Question2' => '2',
15            'Question3' => 'Correct answer'
16        );
17
18        $areAnswersCorrect = true;
19        foreach ($correctAnswers as $question => $answer) {
20            if ($this->post[$question] != $answer) {
21                $areAnswersCorrect = false;
22            }
23        }
24
25        return $areAnswersCorrect;
26    }
27 }

```

ВЫВОДЫ

В ходе лабораторной работы было выполнено создание серверной части сайта на PHP с использованием архитектуры MVC. Для создания удобочитаемых URL использован `UrlRewrite` модуль PHP сервера. Для маршрутизации создан роутер. Валидация параметров формы осуществляется с помощью библиотеки `Valitron`.