

Moldova State University
Faculty of Mathematics and Computer Science

PHP

Аттестационная работа 01.
Разработка веб-приложения для создания и прохождения тестов

Рауса Давид, IAFR 2202

Преподаватель: Nichita Nartea

Chisinau, 2025

Цель

Создать веб-приложение для прохождения теста с возможностью просмотра результатов. Приложение должно предоставлять пользователям интерфейс для прохождения теста, а администратору — доступ к административной панели для просмотра и управления результатами. Вопросы и ответы хранятся в файле или базе данных.

Требования

Функциональные требования

1. Создание и прохождение теста:
 - a. Тема теста на выбор студента.
 - b. Тест содержит вопросы двух типов:
 - i. Один правильный ответ.
 - ii. Несколько правильных ответов.
 - c. Вопросы должны быть отображены с использованием:
 - i. radio — для вопросов с одним правильным ответом.
 - ii. checkbox — для вопросов с несколькими правильными ответами.
 - d. После завершения теста пользователь видит:
 - i. Количество правильных ответов.
 - ii. Процент набранных баллов.
2. Хранение данных:
 - a. Вопросы и ответы тестов могут быть сохранены:
 - i. В файле (например, JSON или текстовом формате).
 - ii. В базе данных.
 - b. Результаты тестов (имя пользователя, ответы на каждый вопрос, набранные баллы) сохраняются и доступны для просмотра в виде таблицы.
3. Аутентификация:
 - a. Доступ к административной панели (dashboard.php) доступен только авторизованным пользователям (администраторам).
 - b. Вход возможен только через аутентификацию с проверкой пароля (регистрация не требуется).

Требования к интерфейсу

1. Главная страница:
 - a. Кнопка "Пройти тест".
2. Страница прохождения теста:
 - a. Отображение вопросов с вариантами ответов:
 - i. Первый вопрос: "Введите ваше имя".
 - ii. Вопросы с несколькими вариантами — с использованием checkbox.
 - iii. Вопросы с одним вариантом — с использованием radio.
3. Результаты теста:
 - a. После завершения теста отображается:
 - i. Количество правильных ответов.
 - ii. Набранные баллы в процентах.
4. Административная панель (dashboard.php):
 - a. Просмотр таблицы с результатами всех пользователей:

- i. Имя пользователя.
- ii. Набранные баллы.
- iii. Время прохождения теста.

Нефункциональные требования

Требования к безопасности

1. Валидируйте все данные, отправляемые пользователем, на стороне сервера.
2. Реализуйте защиту административной панели с помощью аутентификации: тест проходят анонимные пользователи, а результаты просматривают только авторизованные пользователи.

Требования к коду

1. Документируйте код с использованием PHPDoc.
2. Используйте функции или классы для:
 - a. Загрузки и сохранения данных.
 - b. Валидации пользовательских данных.
 - c. Подсчёта результатов теста.
3. Обеспечьте модульность и читаемость кода.

Технологический стек

Язык: PHP

Хранение данных: JSON-файл

Фронтенд: HTML + CSS

Аутентификация: простая форма входа, проверка логина/пароля

Структура проекта

1. project/
 - index.php # Основа
 - test.php # тест
 - submit.php # Обработка и сохранение результатов
 - results.php # результаты (для пользователя)
 - dashboard.php # Админка (только для авторизованных)
 - login.php # входа в админку
 - logout.php # Выход
 - data/
 - test.json # Вопросы и ответы
 - results.json # Результаты теста
 - includes/
 - functions.php # Загрузка/сохранение/валидация
 - auth.php # Аутентификация
 - style.css # Стили

Код:

index.php — главная страница

```
<html>

<head>

  <title>Тестирование</title>

  <link rel="stylesheet" href="style.css">

</head>

<body>

  <h1>Добро пожаловать, товарищ!</h1>

  <a href="test.php" class="button">Пройти тест</a>

  <a href="login.php" class="button">Вход в админку КПСС</a>

</body>

</html>
```

test.php — страница прохождения теста

```
<?php
$data = json_decode(file_get_contents("data/test.json"), true);
?>

<html>
<head>
    <title>Тест</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <h2>Пройти тест</h2>
    <form method="post" action="submit.php">
        <label>Введи имя:</label><br>
        <input type="text" name="username" required><br><br>

        <?php foreach ($data as $index => $question): ?>
            <div>
                <p><?=( $index+1 ) . ". " . $question['question'] ?></p>
                <?php foreach ($question['options'] as $i => $option): ?>
                    <label>
                        <input type="<?= $question['type'] == 'multiple' ? 'checkbox' : 'radio' ?>"
                            name="answer[<?= $index ?>]<?= $question['type'] == 'multiple' ? '[' : " ?>"
                            value="<?= $i ?>"> <?= $option ?>
                    </label><br>
                <?php endforeach; ?>
            </div><br>
        <?php endforeach; ?>

        <button type="submit">Отправить</button>
    </form>
</body></html>
```

[submit.php — обработка результатов](#)

```
<?php
session_start();
require 'includes/functions.php';
$testData = loadTestData();
$username = trim($_POST['username']);
$answers = $_POST['answer'];
$score = 0;
$total = count($testData);
foreach ($testData as $index => $q) {
    $correct = $q['correct'];
    $user = isset($answers[$index]) ? $answers[$index] : [];
    if ($q['type'] === 'single') {
        if ($user == $correct) $score++;
    } else {
        sort($user);
        sort($correct);
        if ($user === $correct) $score++; }
}
$percent = round(($score / $total) * 100);
$entry = [
    'name' => $username,
    'score' => $percent,
    'time' => date("Y-m-d H:i:s"),
    'answers' => $answers
];
saveResult($entry);
$_SESSION['result'] = $entry;
header("Location: results.php");
```

[results.php — вывод результата пользователю](#)

```
<?php
session_start();
$result = $_SESSION['result'] ?? null;
```

```

if (!$result) {
    header("Location: index.php");
    exit;}
?>
<html>
<head>
    <title>Результаты</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <h2>Результат теста</h2>
    <p>Имя: <?= htmlspecialchars($result['name']) ?></p>
    <p>Результат: <?= $result['score'] ?>%</p>
    <a href="index.php">На главную</a>
</body>
</html>

```

[dashboard.php](#) — административная панель

```

<?php
session_start();
require 'includes/auth.php';
requireLogin();
$results = json_decode(file_get_contents("data/results.json"), true);
?>
<html>
<head>
    <title>Админ-панель Ленина</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <h2>Результаты пользователей</h2>
    <table border="1">
        <tr>

```

```

        <th>Имя</th>

        <th>Баллы (%)</th>

        <th>Время</th>

    </tr>

    <?php foreach ($results as $r): ?>

        <tr>

            <td><?= htmlspecialchars($r['name']) ?></td>

            <td><?= $r['score'] ?></td>

            <td><?= $r['time'] ?></td>

        </tr>

    <?php endforeach; ?>

</table>

<a href="logout.php">Выйти из партии</a>

</body>

</html>

```

[login.php — форма входа](#)

```

<?php
session_start();
if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    if ($_POST['password'] === 'admin123') {
        $_SESSION['admin'] = true;
        header("Location: dashboard.php");
        exit;
    } else {
        $error = "Неверный пароль";
    }
}
?>

<html>

<head>

    <title>Вход в партию</title>

    <link rel="stylesheet" href="style.css">

```



```

</head>
<body>
    <h2>Вход в админку</h2>
    <?php if (isset($error)) echo "<p style='color:red;'>$error</p>"; ?>
    <form method="post">
        <label>Пароль администратора:</label><br>
        <input type="password" name="password" required><br><br>
        <button type="submit">Войти</button>
    </form>
</body>
</html>

```

[logout.php — выход из админки](#)

```

<?php
session_start();
session_destroy();
header("Location: index.php");

includes/functions.php

<?php
/**
 * @return array
 */

function loadTestData() {return json_decode(file_get_contents("data/test.json"), true);}

/**
 * @param array $entry
 * @return void
 */

function saveResult(array $entry) {
    $file = "data/results.json";
    $data = file_exists($file) ? json_decode(file_get_contents($file), true) : [];
    $data[] = $entry;
    file_put_contents($file, json_encode($data, JSON_PRETTY_PRINT));}

```

[includes/auth.php](#)

```
<?php
/**
 * @return void
 */
function requireLogin() {
    if (!isset($_SESSION['admin']) || $_SESSION['admin'] !== true) {
        header("Location: login.php");
        exit; }}

```

[data/test.json — тест](#)

```
[
  {
    "question": "Кто пристрелил Романовых?",
    "type": "single",
    "options": ["Путин", " Ленин ", "Гитлер"],
    "correct": "1"
  },
  {
    "question": "Выберите цвета флага Украины",
    "type": "multiple",
    "options": ["Синий", "Зелёный", "Жёлтый"],
    "correct": ["0", "2"]
  }
]
```

[style.css — стили](#)

```
body {
    font-family: Arial, sans-serif;
    margin: 40px;
    background: #f0f0f0;
}
h1, h2 {
    color: #333;
}
```

```

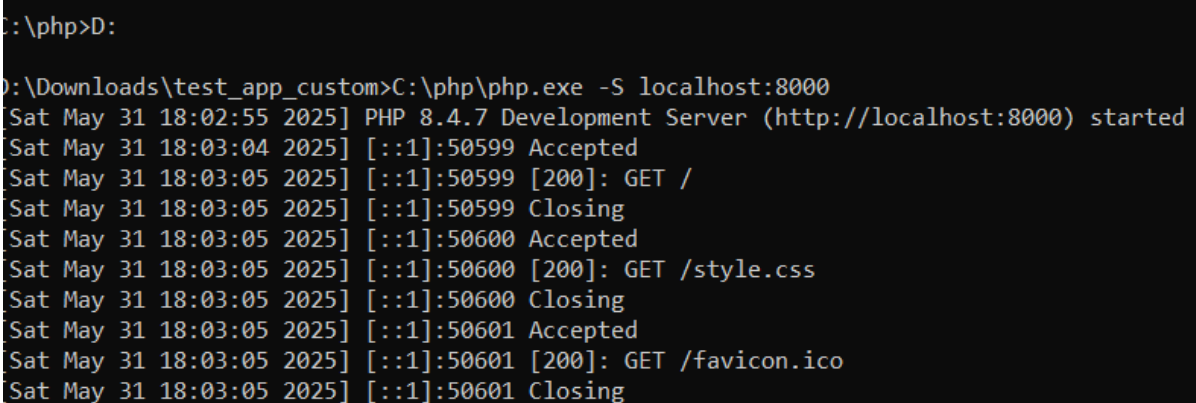
}

.button {
    padding: 10px 20px;
    background: #0056b3;
    color: white;
    text-decoration: none;
    border-radius: 5px;
    margin-right: 10px;
}

form {
    background: #fff;
    padding: 20px;
    border-radius: 8px;
}

```

Вывод:



```

C:\php>D:
D:\Downloads\test_app_custom>C:\php\php.exe -S localhost:8000
[Sat May 31 18:02:55 2025] PHP 8.4.7 Development Server (http://localhost:8000) started
[Sat May 31 18:03:04 2025] [::1]:50599 Accepted
[Sat May 31 18:03:05 2025] [::1]:50599 [200]: GET /
[Sat May 31 18:03:05 2025] [::1]:50599 Closing
[Sat May 31 18:03:05 2025] [::1]:50600 Accepted
[Sat May 31 18:03:05 2025] [::1]:50600 [200]: GET /style.css
[Sat May 31 18:03:05 2025] [::1]:50600 Closing
[Sat May 31 18:03:05 2025] [::1]:50601 Accepted
[Sat May 31 18:03:05 2025] [::1]:50601 [200]: GET /favicon.ico
[Sat May 31 18:03:05 2025] [::1]:50601 Closing

```

Рисунок 1 Запуск проекта

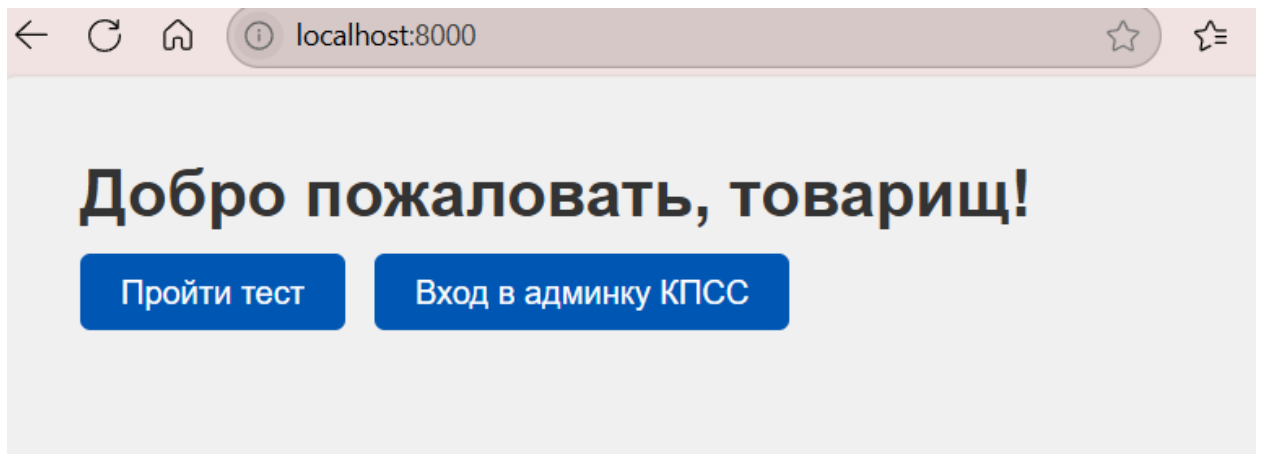


Рисунок 2 Стартовая страница

A screenshot of a web form titled 'Пройти тест' in bold black font. The form is set against a light gray background. It contains a text input field with the label 'Введи имя:' and the value 'Raisa'. Below this, there are two questions. The first question is '1. Кто пристрелил Романовых?' with three radio button options: 'Путин', 'Ленин' (which is selected), and 'Гитлер'. The second question is '2. Выберите цвета флага Украины' with three checkbox options: 'Синий' (checked), 'Зелёный', and 'Жёлтый'. At the bottom of the form is a button labeled 'Отправить'.

Рисунок 3 Тест

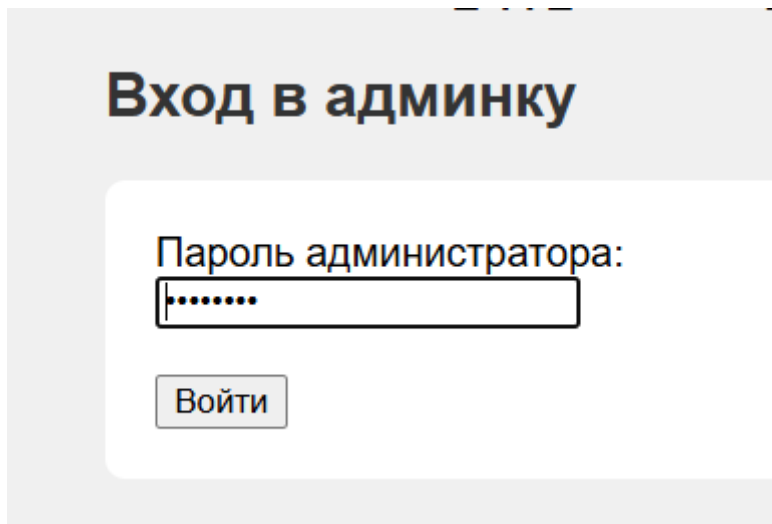


Рисунок 4 Вход в админку

```
D:\Downloads\test_app_custom>C:\php\php.exe -S localhost:8000
[Sat May 31 18:02:55 2025] PHP 8.4.7 Development Server (http://localhost:8000) started
[Sat May 31 18:03:04 2025] [::1]:50599 Accepted
[Sat May 31 18:03:05 2025] [::1]:50599 [200]: GET /
[Sat May 31 18:03:05 2025] [::1]:50599 Closing
[Sat May 31 18:03:05 2025] [::1]:50600 Accepted
[Sat May 31 18:03:05 2025] [::1]:50600 [200]: GET /style.css
[Sat May 31 18:03:05 2025] [::1]:50600 Closing
[Sat May 31 18:03:05 2025] [::1]:50601 Accepted
[Sat May 31 18:03:05 2025] [::1]:50601 [200]: GET /favicon.ico
[Sat May 31 18:03:05 2025] [::1]:50601 Closing
[Sat May 31 18:05:08 2025] [::1]:50617 Accepted
[Sat May 31 18:05:08 2025] [::1]:50617 [200]: GET /test.php
[Sat May 31 18:05:08 2025] [::1]:50617 Closing
[Sat May 31 18:05:08 2025] [::1]:50618 Accepted
[Sat May 31 18:05:08 2025] [::1]:50618 [200]: GET /style.css
[Sat May 31 18:05:08 2025] [::1]:50618 Closing
[Sat May 31 18:05:08 2025] [::1]:50619 Accepted
[Sat May 31 18:05:08 2025] [::1]:50619 [200]: GET /favicon.ico
[Sat May 31 18:05:08 2025] [::1]:50619 Closing
[Sat May 31 18:05:49 2025] [::1]:50621 Accepted
[Sat May 31 18:05:49 2025] [::1]:50621 [200]: POST /submit.php
[Sat May 31 18:05:49 2025] [::1]:50621 Closing
[Sat May 31 18:05:49 2025] [::1]:50622 Accepted
[Sat May 31 18:05:49 2025] [::1]:50622 [200]: GET /favicon.ico
[Sat May 31 18:05:49 2025] [::1]:50622 Closing
[Sat May 31 18:06:02 2025] [::1]:50624 Accepted
[Sat May 31 18:06:02 2025] [::1]:50624 [200]: GET /favicon.ico
[Sat May 31 18:06:02 2025] [::1]:50624 Closing
[Sat May 31 18:06:03 2025] [::1]:50625 Accepted
```

Рисунок 5 Логирование

Заключение

В этой аттестационной работе я сделала простое, но рабочее веб-приложение для прохождения тестов. Пользователь заходит, вводит имя и отвечает на вопросы — а потом сразу видит свой результат в процентах. Всё быстро и понятно.

Для админов есть отдельная страница с таблицей всех результатов. Чтобы туда попасть, нужен пароль. Данные хранятся в обычных JSON-файлах, так что всё просто и прозрачно. Вопросы можно менять как угодно.

Сайт работает без установки сложных серверов — достаточно PHP. И главное: тест может пройти кто угодно.

Библиография

[PHP For Windows: Binaries and sources Releases](#)