

# Реализация взаимодействия алгоритмов программы

Разделить проект на backend и frontend для каждой страницы:

## Students.php:

```
<?php include_once "../includes/parts/header.php"; ?>

<main>
    <h1>Студенты</h1>

    <h2 style="margin-top: 15px;">Добавить студента</h2>
    <form id="add-student-form" class="add_form">
        <input type="text" name="last_name" placeholder="Фамилия"><br>
        <input type="text" name="first_name" placeholder="Имя"><br>
        <input type="text" name="middle_name" placeholder="Отчество"><br>
        <input type="number" name="group_id" placeholder="ID группы"><br>
        <input type="date" name="dismissal_date"><br>
        <button type="submit" class="add-students-btn">Добавить</button>
    </form>

    <h2 style="margin-top: 15px;">Список студентов</h2>
    <table id="students-table">
        <thead>
            <tr>
                <th>ID</th>
                <th>Фамилия</th>
                <th>Имя</th>
                <th>Отчество</th>
                <th>Группа ID</th>
                <th>Дата отчисления</th>
                <th>Действия</th>
            </tr>
        </thead>
        <tbody>
        </tbody>
    </table>
</main>

<script src="../js/jquery-3.7.1.min.js"></script>
<script src="../js/app.js"></script>

<?php include_once "../includes/parts/footer.php"; ?>
```

```
js > app.js > ready() callback > loadStudents
const tbody = $( '#students-table tbody' );

function loadStudents() { ...
}

// Загрузка при открытии страницы
loadStudents();

// Добавление студента
$( '#add-student-form' ).on( 'submit', function ( e ) { ...
});

// Сохранение изменений при нажатии на "Сохранить"
$( document ).on( 'click', '.save-btn', function () { ...
});

// Удаление студента
$( document ).on( 'click', '.delete-btn', function () { ...
});
```

```
student_api.php X discipline_api.php absence_api.php group_api.php program_api.php teach

controllers > api > student_api.php
1 <?php
2 require_once "../includes/parts/connection.php";
3 require_once "../models/Student.php";
4
5 header( header: "Content-Type: application/json" );
6
7 if ( $ _SERVER[ 'REQUEST_METHOD' ] === 'GET' ) {
8     $ items = Student::Get();
9     echo json_encode( value: $ items );
10    exit();
11 }
12
13 if ( $ _SERVER[ 'REQUEST_METHOD' ] === 'POST' ) {
14     $ data = json_decode( json: file_get_contents( filename: 'php://input' ), associative: true );
15
16     if ( isset( $ data[ 'action' ] ) ) {
17         switch ( $ data[ 'action' ] ) {
18             case 'create':
19                 $ student = new Student();
20                 foreach ( $ data as $ key => $ value ) {
21                     if ( property_exists( object_or_class: $ student, property: $ key ) ) {
22                         $ student->{ $ key } = $ value;
23                     }
24                 }
25                 $ student->Insert();
26                 echo json_encode( value: [ 'status' => 'success' ] );
27                 break;
28
29             case 'update':
30                 $ student = new Student();
31                 $ student->student_id = $ data[ 'student_id' ];
32                 foreach ( $ data as $ key => $ value ) {
33                     if ( property_exists( object_or_class: $ student, property: $ key ) ) {
34                         $ student->{ $ key } = $ value;
35                     }
36                 }
37                 $ student->Update();
38                 echo json_encode( value: [ 'status' => 'success' ] );
39                 break;
40
41             case 'delete':
42                 $ student = new Student();
43                 $ student->student_id = $ data[ 'student_id' ];
44                 $ student->Delete();
45                 echo json_encode( value: [ 'status' => 'success' ] );
46                 break;
47         }
48     }
49    exit();
50 }
```

## Disciplines.php:

```
pages > disciplines.php
1 <?php include_once "../includes/parts/header.php"; ?>
2
3 <main>
4 <h1>Дисциплины</h1>
5
6 <h2 style="margin-top: 15px;">Добавить дисциплину</h2>
7 <form id="add-discipline-form" class="add-form">
8 <input type="text" name="discipline name" placeholder="Название дисциплины"><br>
9 <button type="submit" class="add-students-btn">Добавить</button>
10 </form>
11
12 <h2 style="margin-top: 15px;">Список дисциплин</h2>
13 <table id="disciplines-table">
14 <thead>
15 <tr>
16 <th>ID</th>
17 <th>Название дисциплины</th>
18 <th>Действия</th>
19 </tr>
20 </thead>
21 <tbody></tbody>
22 </table>
23 </main>
24
25 <script src=".../js/jquery-3.7.1.min.js"></script>
26 <script src=".../js/app.js"></script>
27
28 <?php include_once "../includes/parts/footer.php"; ?>
```

```
85 // --- Дисциплины ---
86 const tableBodyDisciplines = $('#disciplines-table tbody');
87
88 > function loadDisciplines() { ...
89 }
90
91 // Загрузка при открытии страницы
92 > if ($('#disciplines-table').length > 0) { ...
93 }
94
95 // Добавление дисциплины
96 > $('#add-discipline-form').on('submit', function (e) { ...
97 });
98
99 // Сохранение изменений
100 > $(document).on('click', '.save-btn', function () { ...
101 });
102
103 // Удаление дисциплины
104 > $(document).on('click', '.delete-btn', function () { ...
105 });
```

```
student_api.php discipline_api.php X absence_api.php group_api.php program_api.php tea
controllers > api > discipline_api.php
1 <?php
2 require_once "../includes/parts/connection.php";
3 require_once "../models/Discipline.php";
4
5 header(header: "Content-Type: application/json");
6
7 > if ($_SERVER['REQUEST_METHOD'] === 'GET') {
8     echo json_encode(value: Discipline::Get());
9     exit();
10 }
11
12 > if ($_SERVER['REQUEST_METHOD'] === 'POST') {
13     $data = json_decode(json: file_get_contents(filename: 'php://input'), associative: true);
14
15 > if (isset($data['action'])) {
16 >     switch ($data['action']) {
17 >         case 'create':
18             $item = new Discipline();
19 >             foreach ($data as $key => $value) {
20 >                 if (property_exists(object_or_class: $item, property: $key)) {
21                     $item->{$key} = $value;
22                 }
23             }
24             $item->Insert();
25             echo json_encode(value: ['status' => 'success']);
26             break;
27
28         case 'update':
29             $item = new Discipline();
30             $item->discipline_id = $data['discipline_id'];
31 >             foreach ($data as $key => $value) {
32 >                 if (property_exists(object_or_class: $item, property: $key)) {
33                     $item->{$key} = $value;
34                 }
35             }
36             $item->Update();
37             echo json_encode(value: ['status' => 'success']);
38             break;
39
40         case 'delete':
41             $item = new Discipline();
42             $item->discipline_id = $data['discipline_id'];
43             $item->Delete();
44             echo json_encode(value: ['status' => 'success']);
45             break;
46     }
47     exit();
48 }
```

## Absences.php:

```
<?php include_once "../includes/parts/header.php"; ?>

<main>
<h1>Пропуски занятий</h1>

<h2 style="margin-top: 15px;">Добавить пропуск</h2>
<form id="add-absence-form" class="add_form">
    <input type="number" name="lesson_id" placeholder="ID занятия"><br>
    <input type="number" name="student_id" placeholder="ID студента"><br>
    <input type="number" name="minutes_missed" placeholder="Минут пропущено"><br>
    <input type="text" name="explanatory_note_path" placeholder="Пояснение (путь)"><br>
    <button type="submit" class="add-student-btn">Добавить</button>
</form>

<h2 style="margin-top: 15px;">Список пропусков</h2>
<table id="absences-table">
    <thead>
        <tr>
            <th>ID</th>
            <th>ID занятия</th>
            <th>ID студента</th>
            <th>Минут пропущено</th>
            <th>Пояснение</th>
            <th>Действия</th>
        </tr>
    </thead>
    <tbody>
    </tbody>
</table>
</main>

<script src="../js/jquery-3.7.1.min.js"></script>
<script src="../js/app.js"></script>

<?php include_once "../includes/parts/footer.php"; ?>
```

```
function loadAbsences() { ...
}

// Автозагрузка
if ($('#absences-table').length > 0) { ...
}

// Добавление
$('#add-absence-form').on('submit', function (e) { ...
});

// Обновление
$(document).on('click', '.save-btn', function () { ...
});

// Удаление
$(document).on('click', '.delete-btn', function () { ...
});
});
```

```
student_api.php discipline_api.php absence_api.php X group_api.php program_api.php te
controllers > api > absence_api.php
1 <?php
2 require_once "../includes/parts/connection.php";
3 require_once "../models/Absence.php";
4
5 header(header: "Content-Type: application/json");
6
7 if ($_SERVER['REQUEST_METHOD'] === 'GET') {
8     echo json_encode(value: Absence::Get());
9     exit();
10 }
11
12 if ($_SERVER['REQUEST_METHOD'] === 'POST') {
13     $data = json_decode(json: file_get_contents(filename: 'php://input', associative: true));
14
15     if (isset($data['action'])) {
16         switch ($data['action']) {
17             case 'create':
18                 $item = new Absence();
19                 foreach ($data as $key => $value) {
20                     if (property_exists(object_or_class: $item, property: $key)) {
21                         $item->{$key} = $value;
22                     }
23                 }
24                 $item->Insert();
25                 echo json_encode(value: ['status' => 'success']);
26                 break;
27
28             case 'update':
29                 $item = new Absence();
30                 $item->absence_id = $data['absence_id'];
31                 foreach ($data as $key => $value) {
32                     if (property_exists(object_or_class: $item, property: $key)) {
33                         $item->{$key} = $value;
34                     }
35                 }
36                 $item->Update();
37                 echo json_encode(value: ['status' => 'success']);
38                 break;
39
40             case 'delete':
41                 $item = new Absence();
42                 $item->absence_id = $data['absence_id'];
43                 $item->Delete();
44                 echo json_encode(value: ['status' => 'success']);
45                 break;
46         }
47         exit();
48     }
49 }
```

# StudentGroups.php

```
pages > student_groups.php
1 <?php include_once "../includes/parts/header.php"; ?>
2
3 <main>
4 <h1>Учебные группы</h1>
5
6 <h2 style="margin-top: 15px;">Добавить группу</h2>
7 <form id="add-group-form" class="add_form">
8   <input type="text" name="group_name" placeholder="Название группы"><br>
9   <button type="submit" class="add-group-btn">Добавить</button>
10 </form>
11
12 <h2 style="margin-top: 15px;">Список групп</h2>
13 <table id="groups-table">
14   <thead>
15     <tr>
16       <th>ID</th>
17       <th>Название группы</th>
18       <th>Действия</th>
19     </tr>
20   </thead>
21   <tbody></tbody>
22 </table>
23 </main>
24
25 <script src="../js/jquery-3.7.1.min.js"></script>
26 <script src="../js/app.js"></script>
27
28 <?php include_once "../includes/parts/footer.php"; ?>
```

```
const tableBodyGroups = $('#groups-table tbody');
function loadGroups() { ...
}
if ($('#groups-table').length > 0) { ...
}
$('#add-group-form').on('submit', function (e) { ...
});
$(document).on('click', '.save-btn', function () { ...
});
$(document).on('click', '.delete-btn', function () { ...
});
```

```
student_api.php discipline_api.php absence_api.php group_api.php X program_api.php te
controllers > api > group_api.php
1 <?php
2 require_once "../../includes/parts/connection.php";
3 require_once "../../models/StudentGroup.php";
4
5 header(header: "Content-Type: application/json");
6
7 if ($_SERVER['REQUEST_METHOD'] === 'GET') {
8   echo json_encode(value: StudentGroup::Get());
9   exit();
10 }
11
12 if ($_SERVER['REQUEST_METHOD'] === 'POST') {
13   $data = json_decode(json: file_get_contents(filename: 'php://input'), associative: true);
14
15   if (isset($data['action'])) {
16     switch ($data['action']) {
17       case 'create':
18         $item = new StudentGroup();
19         foreach ($data as $key => $value) {
20           if (property_exists(object_or_class: $item, property: $key)) {
21             $item->{$key} = $value;
22           }
23         }
24         $item->Insert();
25         echo json_encode(value: ['status' => 'success']);
26         break;
27
28       case 'update':
29         $item = new StudentGroup();
30         $item->group_id = $data['group_id'];
31         foreach ($data as $key => $value) {
32           if (property_exists(object_or_class: $item, property: $key)) {
33             $item->{$key} = $value;
34           }
35         }
36         $item->Update();
37         echo json_encode(value: ['status' => 'success']);
38         break;
39
40       case 'delete':
41         $item = new StudentGroup();
42         $item->group_id = $data['group_id'];
43         $item->Delete();
44         echo json_encode(value: ['status' => 'success']);
45         break;
46     }
47   }
48   exit();
49 }
```

## Discipline\_programs.php:

```
1 | <?php include_once "../includes/parts/header.php"; ?>
2 |
3 | <main>
4 | <h1>Программы дисциплин</h1>
5 |
6 | <h2 style="margin-top: 15px;">Добавить программу</h2>
7 | <form id="add-program-form" class="add_form">
8 |   <input type="number" name="discipline_id" placeholder="ID дисциплины"><br>
9 |   <input type="text" name="topic" placeholder="Тема"><br>
10 |   <input type="text" name="lesson_type" placeholder="Тип занятия"><br>
11 |   <input type="number" name="hours" placeholder="Часов"><br>
12 |   <button type="submit" class="add-program-btn">Добавить</button>
13 | </form>
14 |
15 | <h2 style="margin-top: 15px;">Список программ</h2>
16 | <table id="programs-table">
17 |   <thead>
18 |     <tr>
19 |       <th>ID</th>
20 |       <th>ID дисциплины</th>
21 |       <th>Тема</th>
22 |       <th>Тип занятия</th>
23 |       <th>Часов</th>
24 |       <th>Действия</th>
25 |     </tr>
26 |   </thead>
27 |   <tbody></tbody>
28 | </table>
29 | </main>
30 |
31 | <script src="../js/jquery-3.7.1.min.js"></script>
32 | <script src="../js/app.js"></script>
33 |
34 | <?php include_once "../includes/parts/footer.php"; ?>
```

```
286
287
288 const tableBodyPrograms = $('#programs-table tbody');
289
290 function loadPrograms() { ...
291 }
292
293 // Автозагрузка при открытии страницы
294 if ($('#programs-table').length > 0) { ...
295 }
296
297 // Добавление программы
298 $('#add-program-form').on('submit', function (e) { ...
299 });
300
301 // Обновление программы
302 $(document).on('click', '.save-btn', function () { ...
303 });
304
305 // Удаление программы
306 $(document).on('click', '.delete-btn', function () { ...
307 });
308 }
```

```
student_api.php discipline_api.php absence_api.php group_api.php program_api.php X te
controllers > api > program_api.php
1 | <?php
2 | require_once "../includes/parts/connection.php";
3 | require_once "../models/DisciplineProgram.php";
4 |
5 | header(header: "Content-Type: application/json");
6 |
7 | if ($_SERVER['REQUEST_METHOD'] === 'GET') {
8 |   echo json_encode(value: DisciplineProgram::Get());
9 |   exit();
10 | }
11 |
12 | if ($_SERVER['REQUEST_METHOD'] === 'POST') {
13 |   $data = json_decode(json: file_get_contents(filename: 'php://input'), associative: true);
14 |
15 |   if (isset($data['action'])) {
16 |     switch ($data['action']) {
17 |       case 'create':
18 |         $item = new DisciplineProgram();
19 |         foreach ($data as $key => $value) {
20 |           if (property_exists(object_or_class: $item, property: $key)) {
21 |             $item->{$key} = $value;
22 |           }
23 |         }
24 |         $item->Insert();
25 |         echo json_encode(value: ['status' => 'success']);
26 |         break;
27 |
28 |       case 'update':
29 |         $item = new DisciplineProgram();
30 |         $item->program_id = $data['program_id'];
31 |         foreach ($data as $key => $value) {
32 |           if (property_exists(object_or_class: $item, property: $key)) {
33 |             $item->{$key} = $value;
34 |           }
35 |         }
36 |         $item->Update();
37 |         echo json_encode(value: ['status' => 'success']);
38 |         break;
39 |
40 |       case 'delete':
41 |         $item = new DisciplineProgram();
42 |         $item->program_id = $data['program_id'];
43 |         $item->Delete();
44 |         echo json_encode(value: ['status' => 'success']);
45 |         break;
46 |     }
47 |   }
48 |   exit();
49 | }
```

## Teachers.php:

```
pages > teachers.php
1 <?php include_once "../includes/parts/header.php"; ?>
2
3 <main>
4 <h1>Преподаватели</h1>
5
6 <h2 style="margin-top: 15px;">Добавить преподавателя</h2>
7 <form id="add-teacher-form" class="add_form">
8 <input type="text" name="last_name" placeholder="Фамилия"><br>
9 <input type="text" name="first_name" placeholder="Имя"><br>
10 <input type="text" name="middle_name" placeholder="Отчество"><br>
11 <input type="text" name="login" placeholder="Логин"><br>
12 <input type="text" name="password" placeholder="Пароль"><br>
13 <button type="submit" class="add-teachers-btn">Добавить</button>
14 </form>
15
16 <h2 style="margin-top: 15px;">Список преподавателей</h2>
17 <table id="teachers-table">
18 <thead>
19 <tr>
20 <th>ID</th>
21 <th>Фамилия</th>
22 <th>Имя</th>
23 <th>Отчество</th>
24 <th>Логин</th>
25 <th>Действия</th>
26 </tr>
27 </thead>
28 <tbody></tbody>
29 </table>
30 </main>
31
32 <script src="../../js/jquery-3.7.1.min.js"></script>
33 <script src="../../js/app.js"></script>
34
35 <?php include_once "../includes/parts/footer.php"; ?>
```

```
356 const tableBodyTeachers = $('#teachers-table tbody');
357
358 function loadTeachers() { ...
359 }
360
361 // Автозагрузка
362 if ($('#teachers-table').length > 0) { ...
363 }
364
365 // Добавление
366 $('#add-teacher-form').on('submit', function (e) { ...
367 });
368
369 // Обновление
370 $(document).on('click', '.save-btn', function () { ...
371 });
372
373 // Удаление
374 $(document).on('click', '.delete-btn', function () { ...
375 });
376 }
```

```
student_api.php discipline_api.php absence_api.php group_api.php program_api.php teacher_api.php X
controllers > api > teacher_api.php
1 <?php
2 require_once "../../includes/parts/connection.php";
3 require_once "../../models/Teacher.php";
4
5 header(header: "Content-Type: application/json");
6
7 if ($_SERVER['REQUEST_METHOD'] === 'GET') {
8 echo json_encode(value: Teacher::Get());
9 exit();
10 }
11
12 if ($_SERVER['REQUEST_METHOD'] === 'POST') {
13 $data = json_decode(json: file_get_contents(filename: 'php://input'), associative: true);
14
15 if (isset($data['action'])) {
16 switch ($data['action']) {
17 case 'create':
18 $item = new Teacher();
19 foreach ($data as $key => $value) {
20 if (property_exists(object_or_class: $item, property: $key)) {
21 $item->{$key} = $value;
22 }
23 }
24 $item->Insert();
25 echo json_encode(value: ['status' => 'success']);
26 break;
27
28 case 'update':
29 $item = new Teacher();
30 $item->teacher_id = $data['teacher_id'];
31 foreach ($data as $key => $value) {
32 if (property_exists(object_or_class: $item, property: $key)) {
33 $item->{$key} = $value;
34 }
35 }
36 $item->Update();
37 echo json_encode(value: ['status' => 'success']);
38 break;
39
40 case 'delete':
41 $item = new Teacher();
42 $item->teacher_id = $data['teacher_id'];
43 $item->Delete();
44 echo json_encode(value: ['status' => 'success']);
45 break;
46 }
47 exit();
48 }
49 }
```

**Что было сделано:** реализовано AJAX-взаимодействие между фронтендом и бэкендом. Был создан универсальный **app.js**, который работает с несколькими таблицами: **Students**, **Disciplines**, **Absences**, **StudentGroups**, **DisciplinePrograms** и **Teachers**. Все данные подгружаются динамически через **JSON API**, добавление, редактирование и удаление записей происходит без

перезагрузки страницы. В результате все действия выполняются асинхронно, что обеспечивает плавный пользовательский интерфейс и разделение логики между **frontend** и **backend**.