

*Пепельшев Д.И.*  
*студент*  
*Карелова Р.А.*  
*к.п.н., доцент кафедры ИТ*  
*Нишнетагильский технологический институт (филиал) УрФУ*  
*г. Нижний Тагил, Россия*

## **РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ ДЛЯ РАБОТЫ С РАСПИСАНИЕМ ЗАНЯТИЙ В ВУЗЕ**

### ***Аннотация***

*В статье представлен вариант реализации веб-приложения, предназначенного для отображения расписания и являющегося частью автоматизированной системы для работы с расписанием вуза. Кратко описаны работы, осуществляемые на каждом этапе процесса разработки программного средства, указаны применяемые инструменты, приведены примеры программного кода.*

***Ключевые слова:*** веб-приложение, разработка программного обеспечения, php.

*Pepelyshev D.I.*  
*student*  
*Karelova R. A.*  
*Ph. D., associate Professor of the Department of IT*  
*Nizhniy Tagil Technological Institute (branch)UrFU*  
*Nizhny Tagil, Russia*

## **DEVELOPMENT OF A WEB-APPLICATION AS PART OF AN AUTOMATED SYSTEM FOR WORKING WITH THE SCHEDULE THE UNIVERSITY**

### **Abstract**

The article presents a variant of the implementation of a web application as part of an automated system for working with the schedule of the university. This application was designed to display information. There is a description of the work carried out at each stage of the software development process, the tools used are indicated, and examples of program code are given.

**Keywords:** web application, software, software development, php.

Отсутствие универсальных алгоритмов, позволяющих автоматизировать процесс составления расписания в учебном заведении, обуславливает

применение ручного труда для решения этой задачи. Как правило, такое расписание оформляется на бумажном носителе, а затем может быть занесено, например, в Excel-файл для дальнейшего размещения на сайте учебного заведения. Такой способ работы с расписанием является трудоемким и неудобным как для его составителя, так и для конечных пользователей (студентов, преподавателей).

Решением обозначенной проблемы может являться разработка и внедрение программного средства, позволяющего не только автоматизировать, ускорить и упростить составление расписания занятий, но и обеспечить удобство его использования студентами и преподавателями образовательного учреждения.

Первый этап разработки автоматизированной системы заключался в изучении технологий составления расписания занятий в высших учебных заведениях, проведении анализа существующих аналогичных программных средств, выявлении, анализе и спецификации требований к продукту.

В результате разработки требований было определено, что система должна давать следующие возможности пользователям:

- составление расписания занятий в автоматическом режиме;
- возможность составления расписания учебной сессии;
- отображение расписания занятий в электронном виде;
- наличие фильтра для отображения расписания по группам, преподавателям, парам и дисциплинам.

В данной статье представлен вариант реализации веб-приложения для автоматизированной системы отображения и хранения расписания занятий в вузе.

Вторым этапом разработки веб-приложения для составления и хранения расписания являлось проектирование, во время которого была составлена схема общей архитектуры создаваемого приложения, спроектирована структура базы данных, составлены и написаны основные алгоритмы внутренней работы продукта. Также на данном этапе было произведено эскизное проектирование основных страниц приложения.

На этапе проектирования программного средства было принято решение по использованию клиент-серверной архитектуры. На схеме, предложенной на рисунке 1, видно, что веб-приложение обменивается данными с другим сервером посредством использования API, а по протоколу прикладного уровня передачи данных HTTP - с клиентами приложения, также данный сервер производит прямой обмен данными с базой данных.



Рисунок 1 – Место веб-приложения в схеме архитектуры автоматизированной системы работы с расписанием

На этапе проектирования приложения были составлены алгоритмы, описывающие внутренние принципы работы данного продукта. К таким алгоритмам можно отнести алгоритм, приведенный на рисунке 2, описывающий принцип работы по автоматической расстановке «карточек» выбранного расписания, и алгоритм, изображенный на рисунке 3, показывающий процесс сброса расстановки всех «карточек» для указанного расписания.

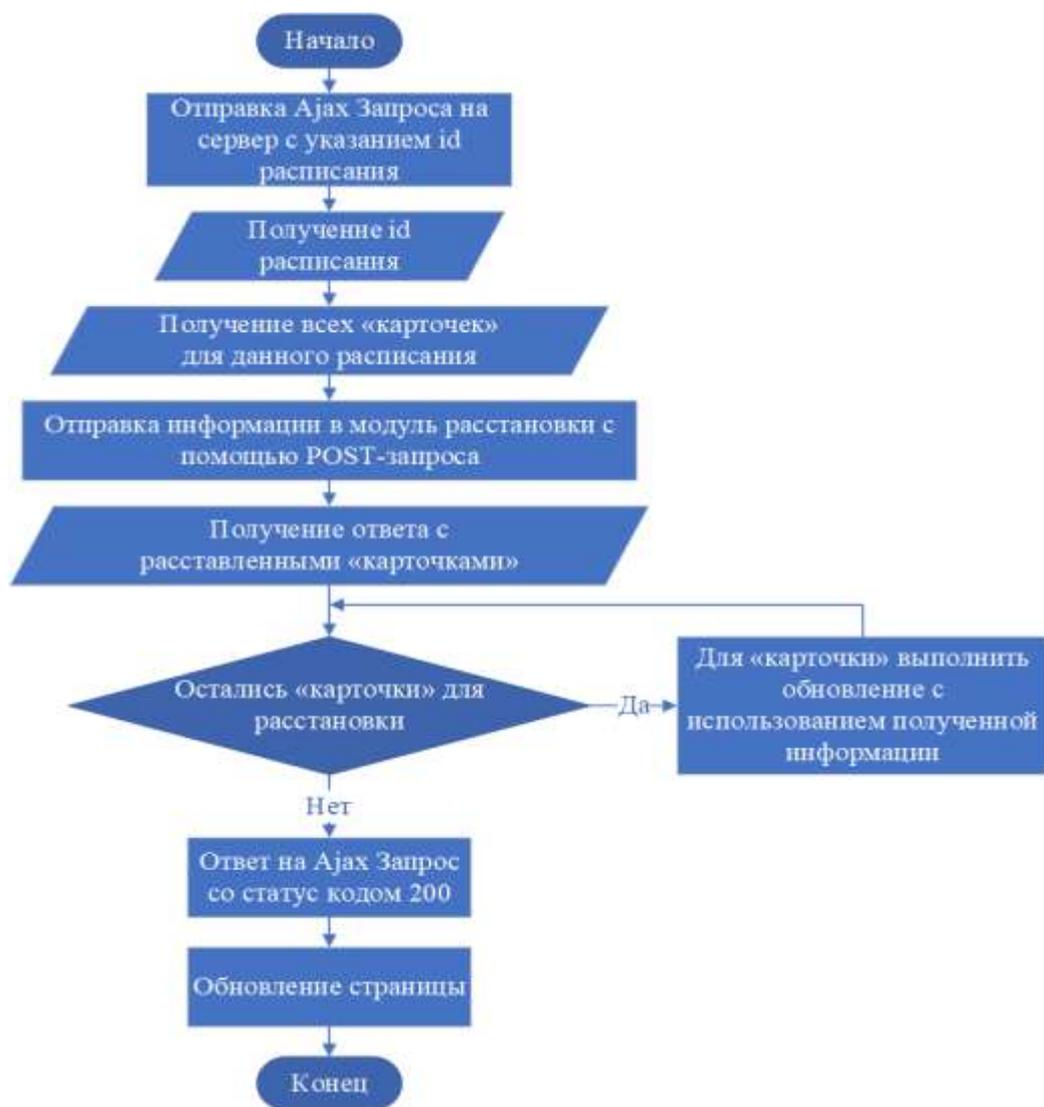


Рисунок 2 – Алгоритм расстановки всех «карточек» выбранного расписания

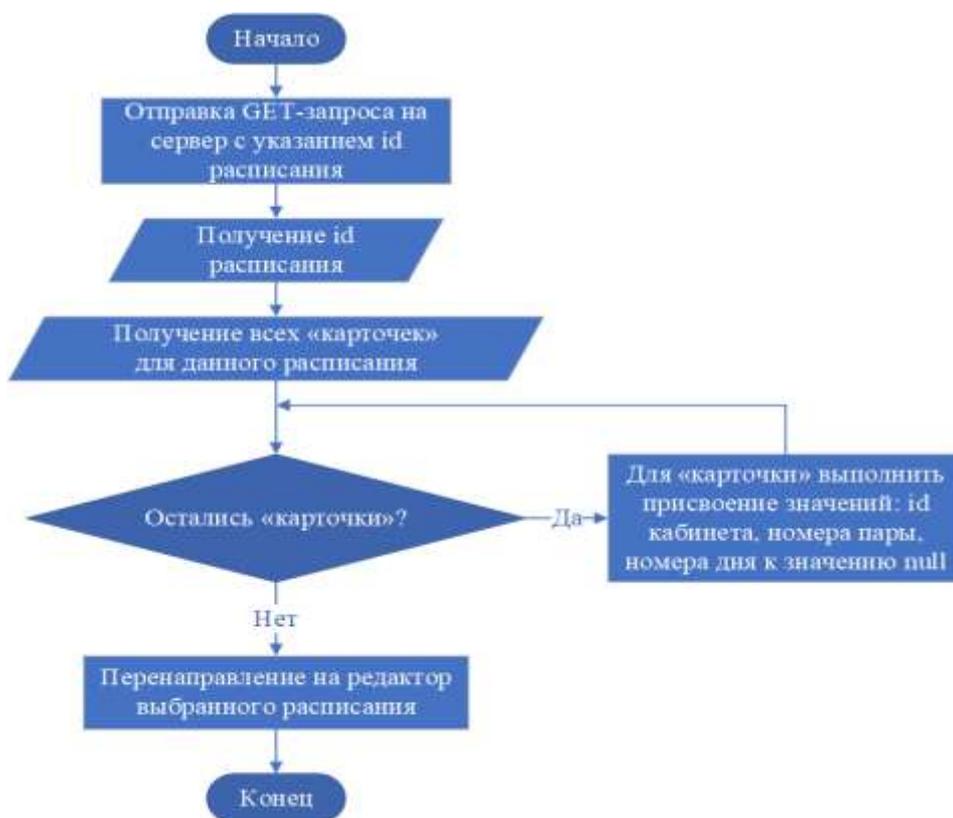


Рисунок 3 – Алгоритм сброса всех «карточек» для выбранного расписания

На этапе проектирования интерфейса веб-приложения были разработаны эскизы главной страницы, предназначенной для отображения расписания для выбранной группы или преподавателя (рисунок 4), и страницы панели администратора, предназначенной для формирования расписания занятий администратором приложения (рисунок 5). Данные эскизы были выполнены с применением основных цветов, шрифтов и стилей оформления в соответствии с правилами, изложенными в брендбуке вуза.

Расписание НТИ (филиал) УрФУ

Выбор группы: T-100801-НТ

Выбор расписания (просмотр архива): Случайно сгенерированное расписание: 60b5e06c827d7

Сегодня 5.6.2021, Числитель

ВТОРНИК

<p>Установки очистки сточных вод и промышленных газов</p> <p>1 Лабораторная работа 331 НУК Долженкова Елена Владимировна Поток №1</p>	<p>Установки очистки сточных вод и промышленных газов</p> <p>Лабораторная работа 331 НУК Долженкова Елена Владимировна Поток №1</p>	<p>Теплоэнергетическое хозяйство металлургических заводов</p> <p>4 Лекции 406 НУК Грузман Вячеслав Моисеевич Поток №1</p>
<p>Установки очистки сточных вод и промышленных газов</p> <p>Лабораторная работа 331 НУК Долженкова Елена Владимировна Поток №1</p>		<p>Технологии Semantic Web</p> <p>6 Лабораторная работа 355 НУК Киреева Наталья Евгеньевна Поток №1</p>

Рисунок 4 – Эскиз главной страницы веб-приложения

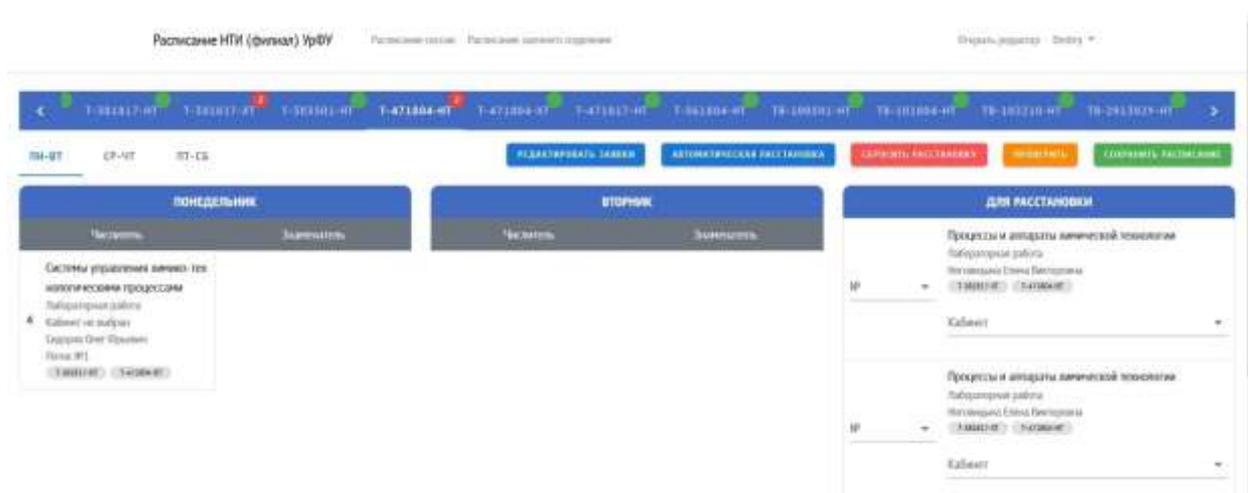


Рисунок 5 – Эскиз панели администрирования для составления расписания

Следующим шагом разработки была непосредственно реализация программного средства. В результате сравнительного анализа возможных средств реализации проектных решений было принято решение о разработке пользовательского интерфейса с использованием фреймворка Vue.js, а серверной части приложения с помощью веб-фреймворка PHP Laravel.

Из особенностей реализации можно отметить рассмотренные ранее алгоритмы, приведенные в листинге 1 и листинге 2.

Листинг 1 – Метод для автоматической расстановки «карточек» расписания

```
<?php
public function arrangement(Schedule $schedule)
{
    $all_cards = $schedule->cards;
    $subject_request = [];
    $locked_lessons = [];
    foreach ($all_cards as $card) {
        $days = json_decode($card['lesson_request']['available
_days']);
        for ($i = 0; $i < count($days); $i++) {
            $days[$i] -= 1;
        }
        $classrooms = Classroom::find(json_decode($card['avail
able_classrooms']));
        $classroom_spited = [[], []];
        foreach ($classrooms as $c) {
            array_push($classroom_spited[$c['building']]['id']
- 1], $c['id']);
        }
        array_push($subject_request, array(
            'id' => $card['id'],
```

```

        'professor' => $card['lesson_request']['teacher_id'],
        'complexity' => $card['lesson_request']['complexit
y'],
        'days' => $days,
        'groups' => json_decode($card['student_groups']),
        'classrooms' => $classroom_spited,
    ));
    if (isset($card['number']) && isset($card['day_type_id
'])) {
        array_push($locked_lessons, array(
            'address' => $this->to_address($card['day_type_id'], $card['number']),
            'subject_request_id' => $card['id']
        ));
    }
    $send_data = array(
        'subject_requests' => $subject_request,
        'locked_lessons' => $locked_lessons
    );
    $response = Http::timeout(60)->post('http://127.0.0.1:9304/makeSchedule', $send_data);
    foreach ($response->json() as $item) {
        $card = $all_cards->find($item['subject_request_id']);
        $from_address = $this->from_address($item['address']);
        $card->number = $from_address['number'];
        $card->day_type_id = $from_address['day_type_id'];
        if ($item['classroom'] > 0) {
            $card->classroom_id = $item['classroom'];
        }
        $card->save();
    }
    return 'ok';
}
}

```

Листинг 2 - Метод для сброса «карточек» расписания

```

<?php
    public function reset_card(Schedule $schedule){
        $lesson_requests = DB::table('lesson_requests')->where('schedule_id', $schedule['id'])->get('id')->toArray();
        $les_req_ids=[];
        foreach ($lesson_requests as $lq){
            array_push($les_req_ids,$lq->id);
        }
        Card::whereIn('lesson_request_id',$les_req_ids)->update([
            'number'=>null,
            'classroom_id'=>null,
            'day_type_id'=>null
        ]);
        return redirect()->route('schedule_editor',['schedule'=>$schedule['id']]);
    }
}

```

Следует отметить, что при разработке приложения было уделено внимание обеспечению безопасности разрабатываемой системы, а именно: реализован процесс аутентификации и авторизации пользователя для получения прав доступа к панели администрирования, предусмотрено шифрование паролей пользователей в БД с использованием алгоритма криптографического хеширования BCrypt, предусмотрена возможность сброса и восстановления пароля по E-Mail адресу, реализована проверка роли пользователя при осуществлении входа, организована безопасная передача данных между клиентом и сервером с помощью использования протокола HTTPS.

Заключительным этапом разработки являлось проведение тестирования, во время которого были составлены и выполнены тесты, разработанные с помощью фреймворка PHPUnit, предназначенного для проведения модульных и интеграционных тестов для продуктов, написанных с использованием языка программирования PHP.

Результатом данной работы стало веб-приложение автоматизированной системы отображения и хранения расписания занятий вуза. Приложение предназначено для обеспечения удобной работы студентов и преподавателей с электронным расписанием занятий, а во взаимодействии с модулем автоматизированного составления расписания упрощает работы по составлению плана занятий для диспетчера данного учреждения.