Оглавление

[Введение 2](#_Toc164201360)

[Модель базы данных сайта представлена ниже 4](#_Toc164201361)

[Дизайн страниц: 6](#_Toc164201362)

[3 Реализация WEB приложения 14](#_Toc164201363)

[Заключение 27](#_Toc164201364)

# Введение

Стандартное расписание АлтГТУ предоставляет информацию о расписание выбранной группы примерно на 2 недели. Предметы (пары) выводятся простым списком (например указаны первая и четвертая пара, записи выведены последовательно, визуализация окна отсутствует); пары идущие в одно время (например предмет ведут два преподавателя) тоже выводятся простым списком, что может затруднить визуальную оценку количества пар. Также одновременно можно просматривать расписание только одной группы. Форма выводимой информации может быть значительно улучшена. В связи с этим появляется задача разработать web-приложение способное убрать указанные недостатки а именно:

* Возможность узнать расписание группы, преподавателя, кабинета (в дальнейшем - разделы).
* Возможность добавить на главный экран несколько разделов, для их сравнения между собой.
* Объединение уроков по времени, а также по другим признакам (урок, группа, кабинет, преподаватель) (опционально).
* Возможность скрывать лишнюю информацию, столбцы.

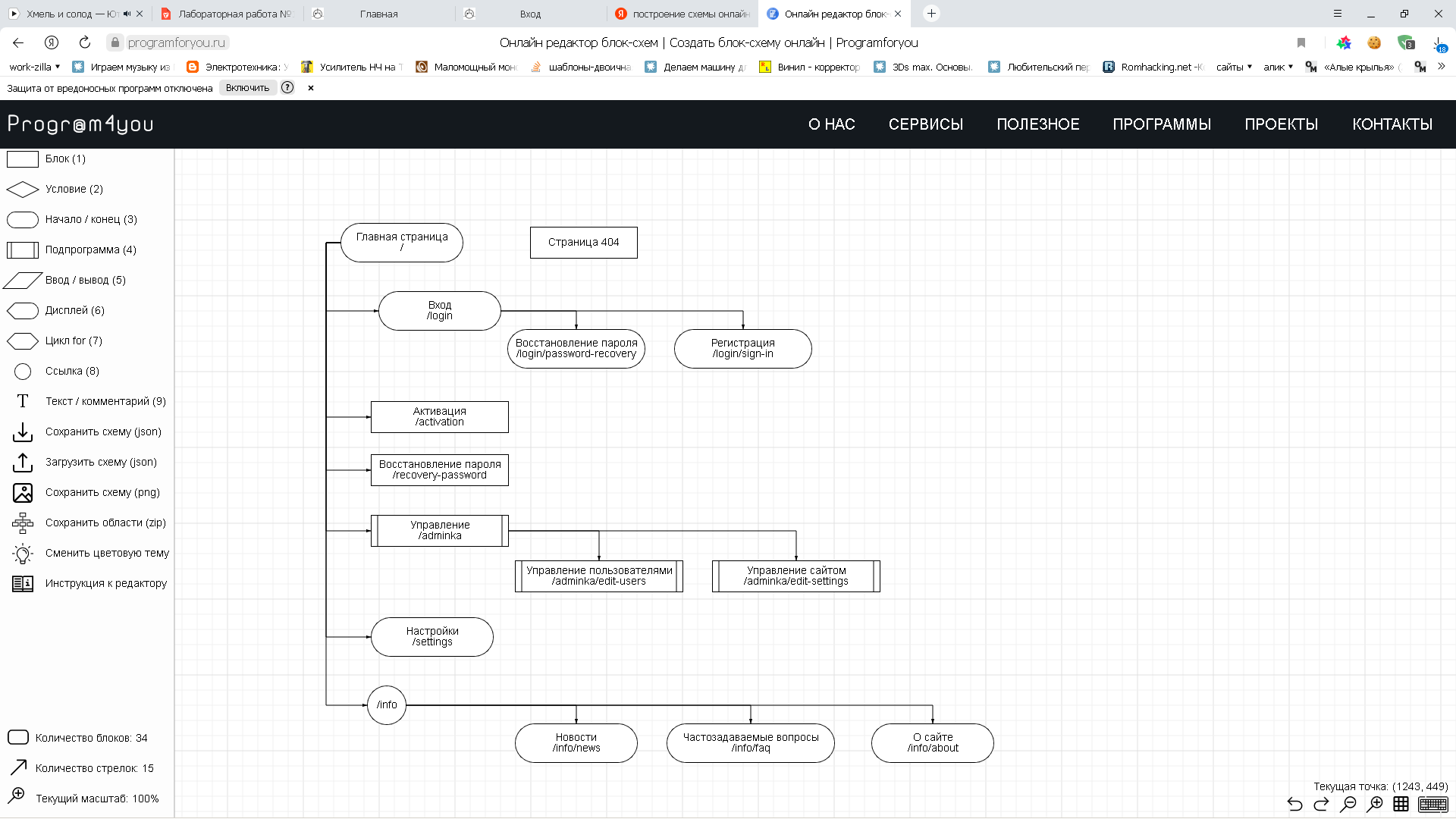


Рисунок 4 – Карта сайта

Сайт имеет стилистику минимального содержания графических элементов, чтобы не отвлекать пользователя. Поэтому вместо картинок используются символы UTF-16

# Модель базы данных сайта представлена ниже

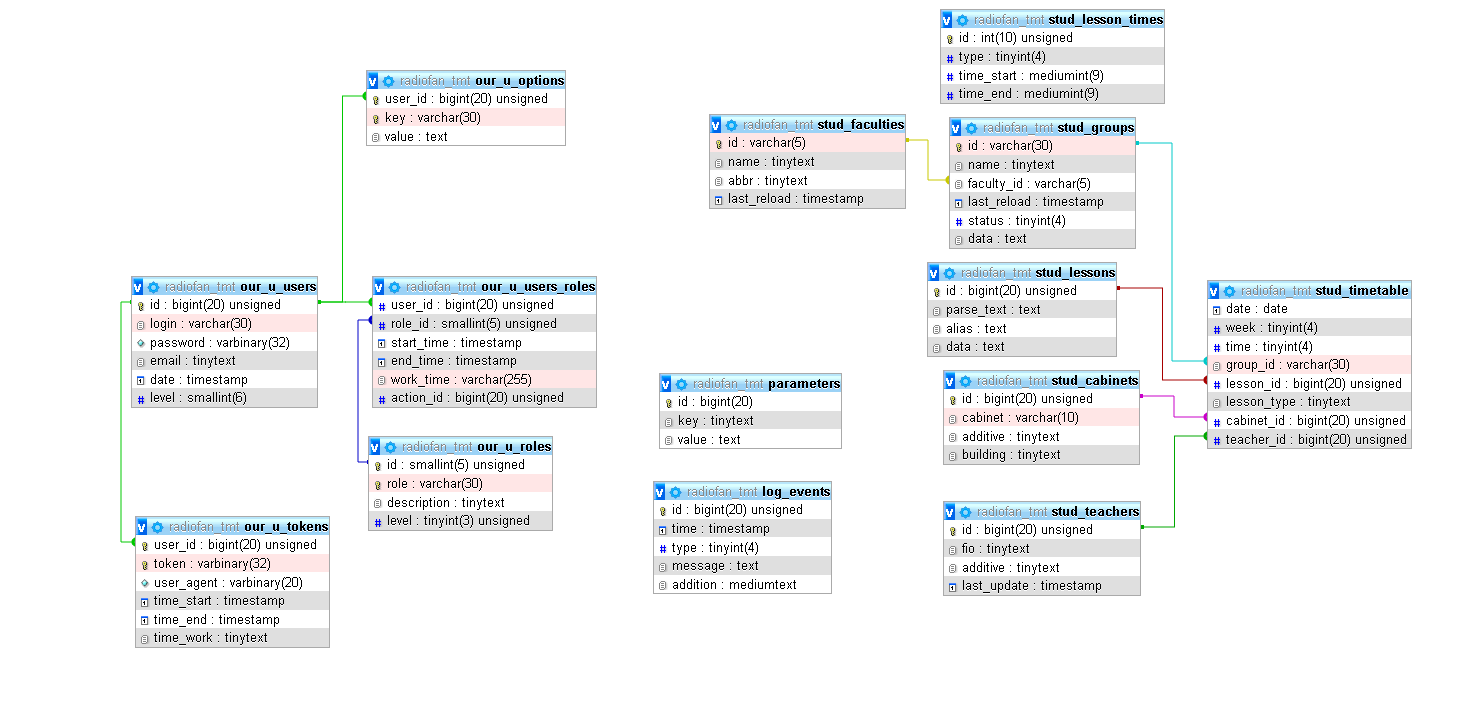


Рисунок 6 – Модель БД сайта

Модель можно разделить условно на три части

- Данные связанные с пользователями (u\_users, u\_options, u\_users\_roles, u\_roles, u\_tokens)

- Данные связанные с расписанием (stud\_timetable, stud\_teachers, stud\_cabinets, stud\_lessons, stud\_groups, stud\_lesson\_times, stud\_faculties)

- Другие данные (parameters, log\_events)

Описание таблиц

u\_users - содержит данные о зарегистрированных пользователях, а именно данные для входа, ID, дату регистрации, уровень пользователя.

u\_options - Key - Value хранилище параметров для каждого пользователя

u\_users\_roles - содержит данные о дополнительных правах пользователя (связь ID юзера, ID права), время вступления в силу, время работы права и время окончания работы права.

u\_roles - содержит имена и описания дополнительных прав

u\_tokens - содержит токены зарегистрированных пользователей, дату старта сессии, дату окончания сессии, время работы сессии.

stud\_timetable - содержит данные об одном уроке, а именно дату занятия, номер недели, номер пары из таблицы stud\_lesson\_times, id группы, урока, кабинета, учителя; тип урока

stud\_teachers - содержит данные о преподавателях, время внесения в базу

stud\_cabinets - содержит данные о кабинетах, корпус, в котором они расположены

stud\_lessons - содержит данные о уроках

stud\_groups - содержит данные о группах, а именно id группы, наименование, id факультета, время последнего парсинга расписания группы, статус существования группы, доп. данные.

stud\_lesson\_times - содержит данные о номерах пар, времени их начала и конца

stud\_faculties - содержит данные о факультетах, а именно id, наименование, аббревиатура, дата последнего обновления данных о наличии групп

parameters - Key - Value хранилище параметров сайта

log\_events - содержит данные лога, дату события, тип и сообщение

# Дизайн страниц:

Для верстки сайта был использована библиотека bootstrap 3.

Каждая страница имеет шапку (header) и подвал (footer). Шапка содержит раздел head, логотип ссылку на главную страницу и набор элементов меню. Содержимое меню зависит от текущей страницы и от свойств текущего пользователя. Подвал содержит скрытое стандартное всплывающее окно.



Рисунок 7 – Сверстанная шапка с меню

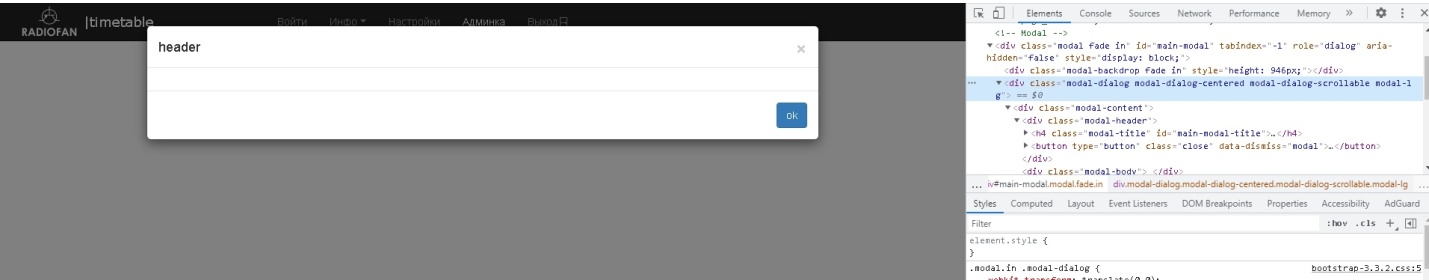


Рисунок 8 – Стандартное всплывающее окно

Главная страница состоит из 2 частей

Часть управления:

Содержит три вкладки: настройки, добавить раздел (открывает всплывающее окно с диалогом добавления), управлять разделами.



Рисунок 9 – Часть управления

Вкладка настройки содержит различные переключатели и другие элементы.

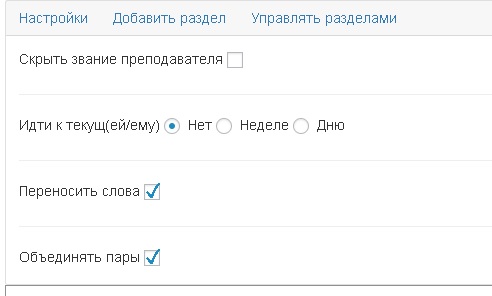


Рисунок 10 – Вкладка настройки

Окно с диалогом добавления содержит три кнопки с выбором типа раздела, текстовый поле ввода с поиском, для выбора конкретного элемента, поле ввода названия раздела, кнопку добавления.

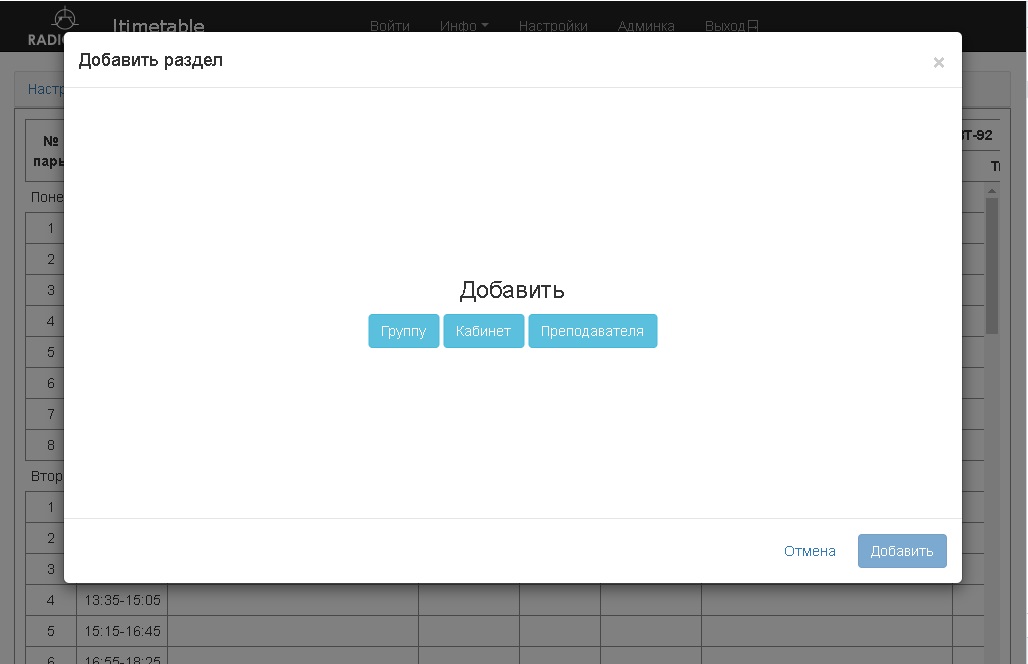


Рисунок 11 – Диалог добавления раздела

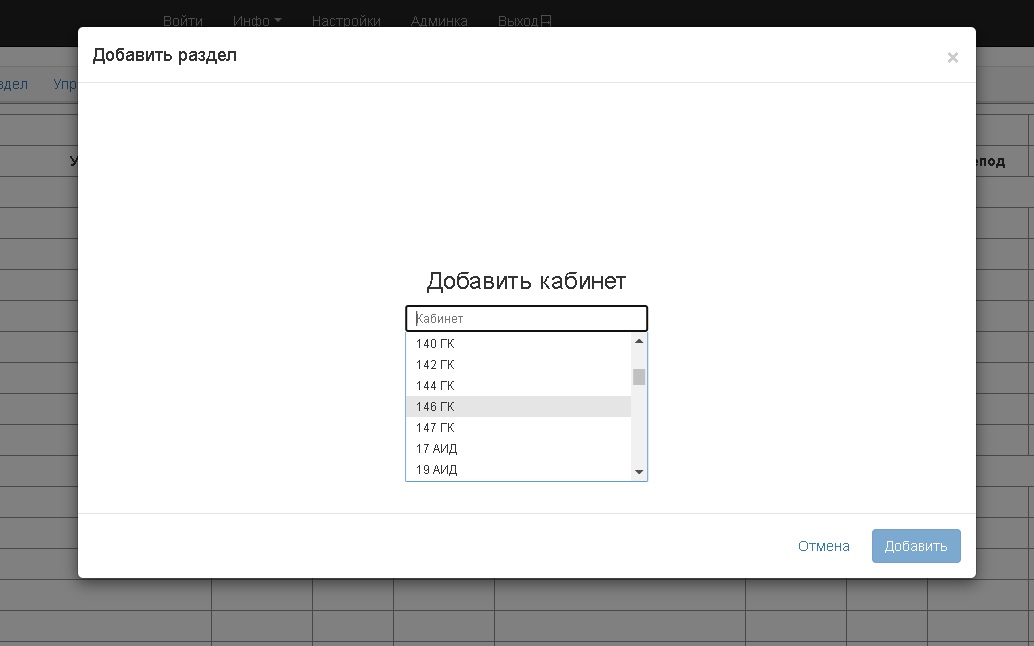


Рисунок 12 – Диалог добавления раздела

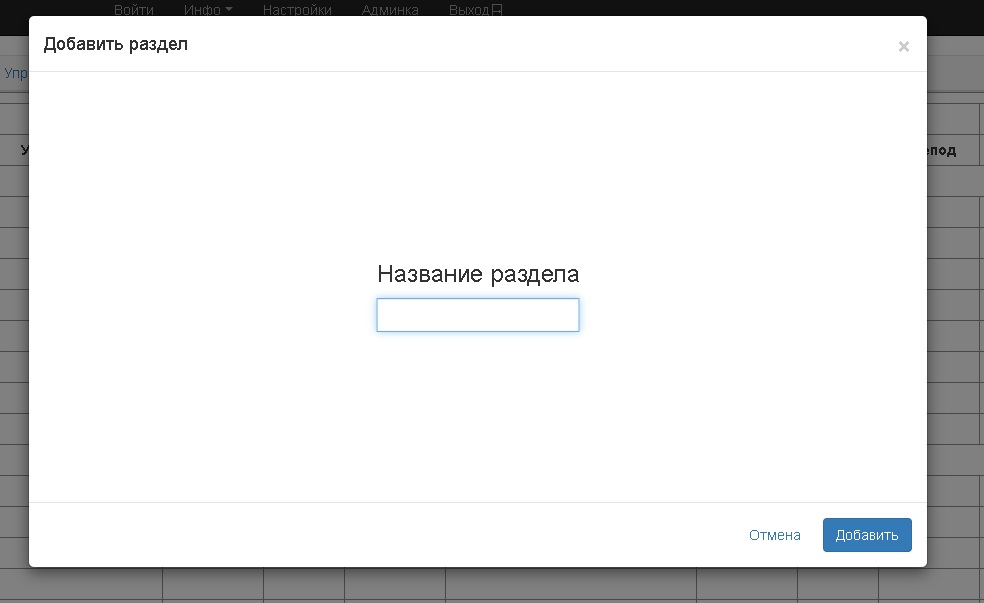


Рисунок 13 – Диалог добавления раздела

Вкладка управления разделами содержит плашки с добавленными разделами с возможностью управлять ими (переносить местами, включать/выключать столбцы, менять название, удалять разделы)

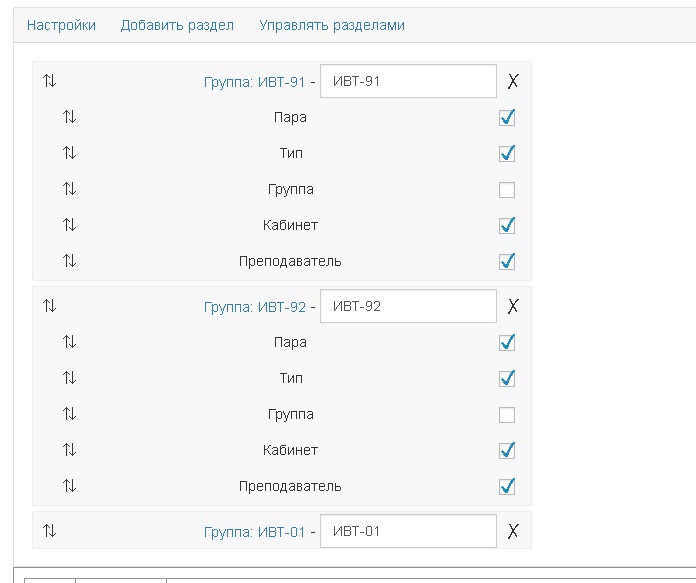


Рисунок 14 – Вкладка управления разделами

Часть таблицы:

Содержит таблицу с расписанием указанных разделов (заполняется из БД)

Шапка и 2 левых столбца являются плавающими.

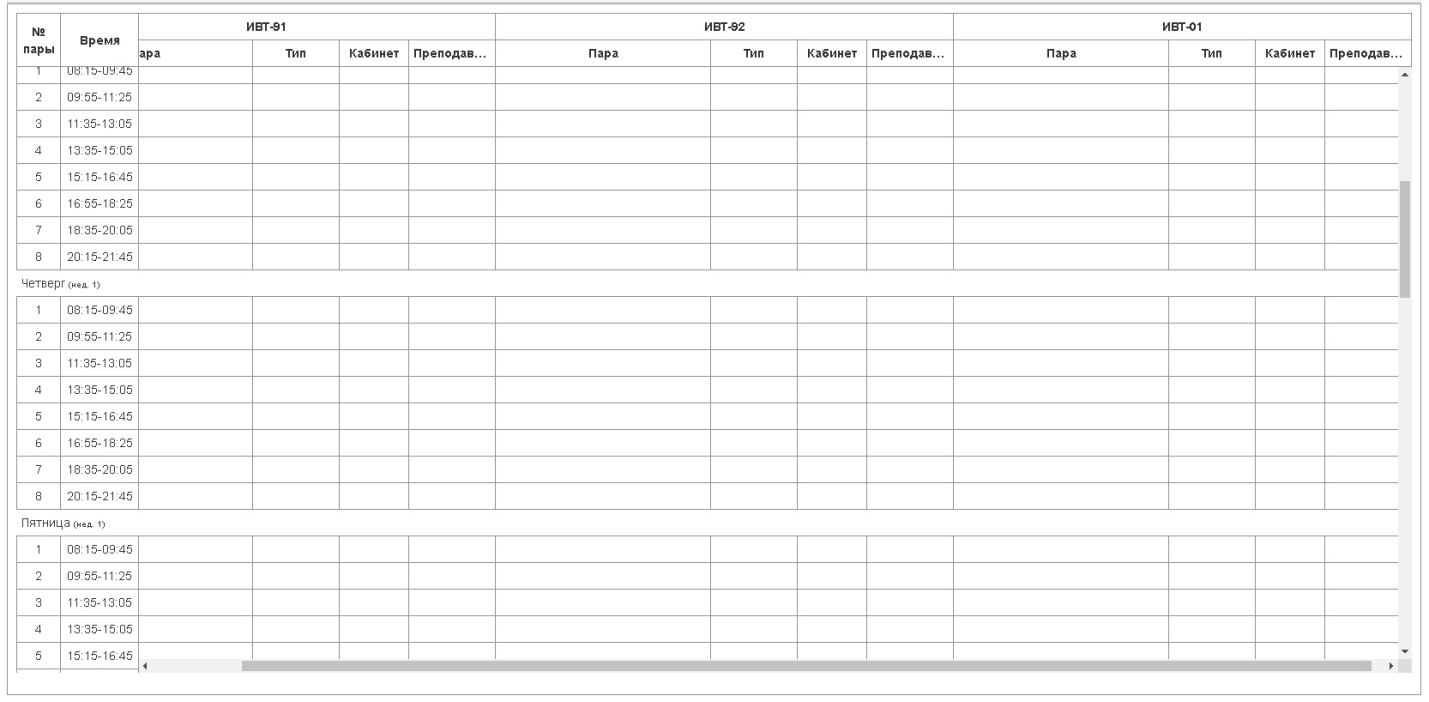


Рисунок 15 – Таблица

Страница входа/регистрации содержит 3 формы: входа, регистрации и восстановления пароля.

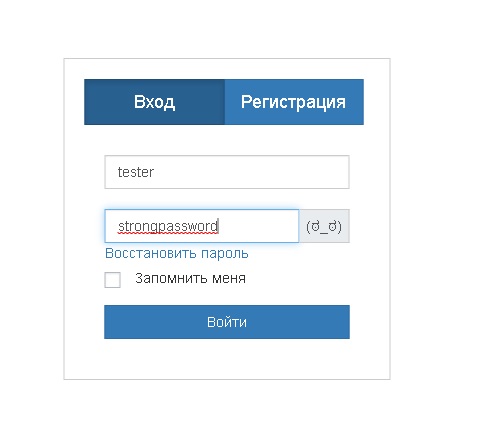


Рисунок 16 – Форма входа



Рисунок 17 – Форма регистрации

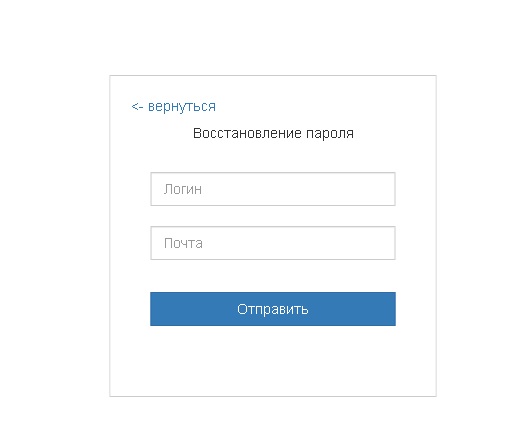


Рисунок 18 – Форма восстановления пароля

Так же имеются три информационные страницы содержащие простую текстовую информацию.

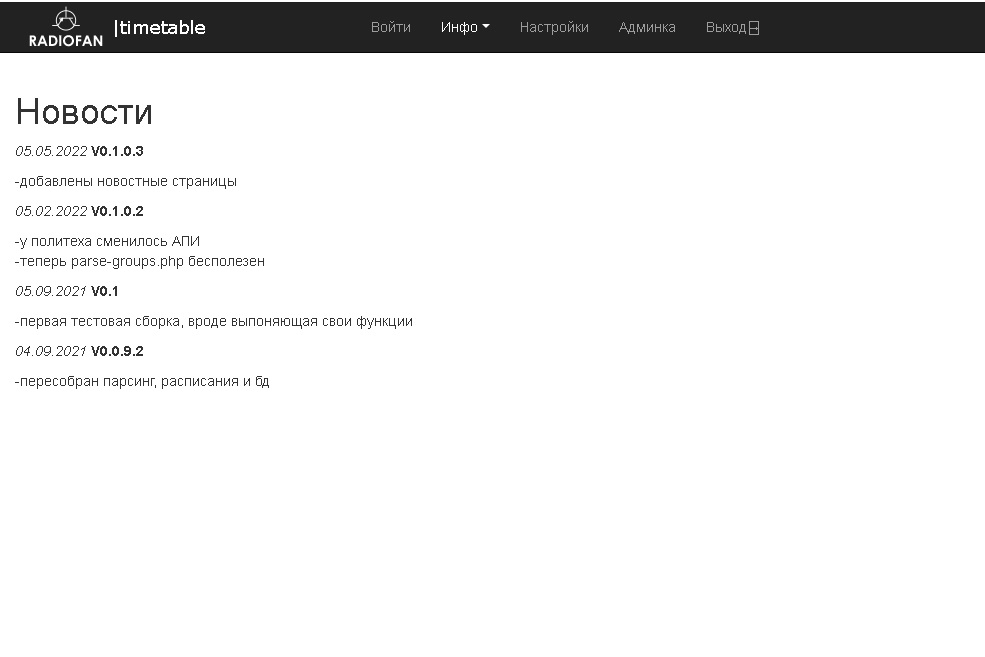


Рисунок 19 – Страница Новости

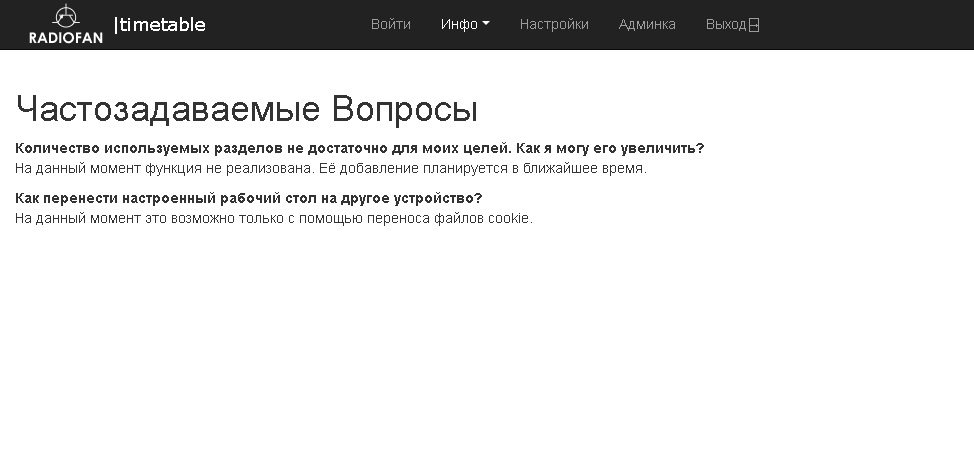


Рисунок 20 – Страница Частозадаваемые Вопросы



Рисунок 21 – Страница О сайте

# 3 Реализация WEB приложения

Работу сайта можно условно разделить на несколько групп:

- Cron события

- Обработка запросов

- Логика объекта пользователя

- Логика расписания

- Вывод страниц

- Дополнительные функции

Cron события:

Скрипты запускаются в определенное время, через определенные интервалы.

Папка со скриптами: www/includes/cron/

Доступ извне в папку запрещен (.htaccess)

Каждый скрипт должен подключать init.php (инициализирует ядро, исключая создание экземпляра пользователя и других моментов)

parse-groups.php - производит парсинг групп и факультетов с помощью найденного API (www.altstu.ru/main/schedule/ws/group/?f=<faculti id>). На данный момент API отключено.

parse-timetable.php - производит парсинг расписания групп. Запускается каждые 12 часов.

Также к этим скриптам относится файл www/includes/functions/parse-functions.php содержащий функции парсинга html файлов и добавления данных в БД.

Обработка запросов:

Сайт имеет возможность обрабатывать запросы (события) пользователей на изменение или получение каких-либо данных.

Папка со скриптами www/includes/actions/

События подключаются автоматически (файлы оканчивающиеся на \*-actions.php).

actions.php - подключает файлы обработчиков событий, запускает обработку событий при наличии $\_REQUEST['action'] с указанным событием.

login-actions.php - событие входа пользователя, событие регистрации пользователя, проверка незанятости логина, запрос на восстановление пароля, событие выхода пользователя.

user-settings-actions.php - событие восстановления пароля

Логика объекта пользователя:

Ядро автоматически создает объект текущего пользователя (rad\_user), если точка запуска index.php или ajax.php.

Папка со скриптами www/includes/classes/user-class/

Пользователь имеет два состояния - гость или зарегистрированный пользователь.

Гость имеет пустые данные.

Данные пользователя - id, логин, хэш пароля, дата регистрации, почта (таблица u\_users), параметры (сериализованные данные относящиеся к пользователю, таблица u\_options), роли (уровень пользователя, таблица u\_users; доступные пользователю возможности, таблица u\_users\_roles).

Класс пользователя разделен на подклассы выполняющие свои функции.

01-user-base.php - основа класса, предоставляет геттеры/сеттеры базовых данных пользователя, загружает экземпляр пользователя, создает роли и параметры.

01-1-user-roles.php - класс отвечающий за возможности пользователя. Имеет методы добавления/удаления ролей, временных ролей пользователю, метод проверки возможности выполнения роли пользователем.

01-2-user-options.php - имеет методы добавление/чтение/удаление параметров текущего пользователя, их кэширования на время выполнения скриптов сайта.

02-user-auth.php - отвечает за вход пользователя на сайт. Вход на сайт происходит с помощью пары логин-пароль, дальнейшее запоминание пользователя происходит с помощью токена сессионного/длительного хранения в cookie пользователя. Класс имеет методы работы с токенами (генерация, удаление, парсинг, таблица u\_tokens), входа по логину-паролю, входа по токену, выход пользователя, проверки занятости логина.

03-user-class.php - финальный класс, загружает текущего пользователя, также имеет методы создание нового пользователя, смены пароля и верификации почты.

Логика расписания:

Основная логика реализована в парсинге данных с помощью cron событий, и вывода данных расписания в удобочитаемом формате в зависимости от настроек хранящихся в cookie пользователя.

www/includes/classes/cookie-validator-class.php - отвечает за валидацию настроек в cookie и их пересохранение, а именно:

выбранные разделы (не больше MAX\_PARTS\_TIMETABLE (www/defines.php)), их тип, id, название, сортировка столбцов, их размер и видимость. Также настройки: скрывать звание преподавателя, перенос слов, идти к текущему нет/неделе/дню, объединять пары.

www/themes/default/index.php - отвечает за вывод тела страницы с таблицей расписания. После парсинга настроек из cookie загружаются данные из БД, обрабатываются, и выводятся в html формате.

Имеется возможность одновременно выводить расписание нескольких групп, кабинетов, преподавателей одновременно, смена местами и скрытие ненужных столбцов (название пары, тип пары, преподаватель, кабинет, название группы). С помощью настроек включить автоматический переход к текущему дню или неделе, также возможность объединения пар по их номерам (всегда отображается 8 строк в каждом дне) или выключить их объединение (каждая пара имеет свою отдельную строку, аналогия расписания АлтГТУ).

Вывод страниц:

Выбор страниц основывается на текущем URL. Сайт поддерживает ЧПУ, их работа представлена классом rad\_url (www/includes/classes/url-class.php). Данный класс предоставляет методы создания узлов URL с уникальными путями и id, создания параметров в URL (node/parameter1/parameter2/), парсинг текущего URL, получение id текущей страницы, параметров, хлебных крошек.

За вывод текущей страницы отвечает класс rad\_pages\_viewer (www/includes/classes/pages-class.php). Класс регистрирует доступные страницы, задает их id, заголовок, уровень доступа по умолчанию, доступность по ролям, файл с функциями вывода и проверок (www/themes), регистрирует узлы URL. Затем производит выбор текущей страницы, подгрузку файла с функциями её отображения, проверяет разрешение на отображение текущей страницы, затем подготовка данных для подвала и шапки, и затем вывод целой страницы.

Дополнительные функции:

Ядро имеет ряд функций, которые могут использоваться в разных частях сайта: логгер (www/includes/classes/log-class.php) - логирует ошибки php, обертка над cURL (www/includes/classes/parser-class.php) - производит запросы к внешним страницам и обрабатывает их ответы, класс для работы с БД (www/includes/classes/db-class.php), глобальные настройки сайта (www/includes/classes/data-class.php**),** вывод уведомлений для пользователей (www/includes/classes/alerts-class.php), файлы переводов (www/langs/) - файлы с языковыми константами, файл с техническими константами (www/defines.php), точки входа ajax запросов (ответ - json данные) и запросов к сайту (ответ - html страница) (www/ajax.php, www/index.php), настройки сервера через файл htaccess (www/.htaccess) - RewriteRule для работы ЧПУ.

Для работы интерактивных элементов на сайте использовались скрипты написанные на JavaScript. Основой была выбрана библиотека jQuery (www/libs/jquery-3.4.1.min.js) и скрипты Bootstrap (www/libs/bootstrap-3.3.2.min.js).

Для работы перетаскивания разделов и столбцов во вкладке "Управлять разделами" использована библиотека jQuery UI Sortable (www/libs/jquery-ui-1.12.1.sortable.min.js).

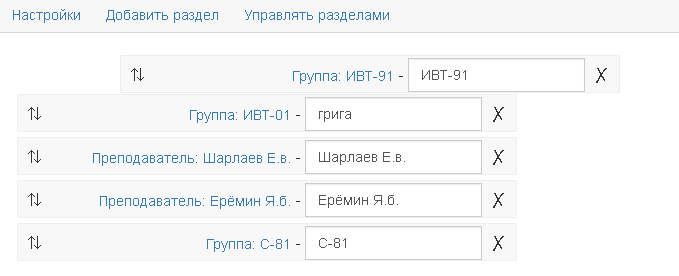


Рисунок 22 – Перетаскивание разделов

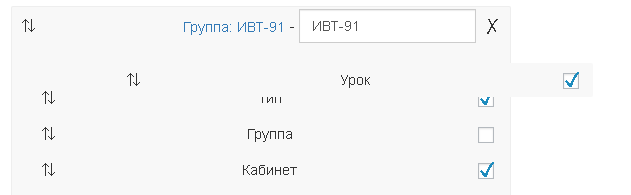


Рисунок 23 – Перетаскивание столбцов

Для реализации поля ввода с подсказками была разработана библиотека на основе jQuery (www/libs/jquery.searchSelect.js).



Рисунок 24 – Поле ввода с выпадающим списком подсказок

Основные настройки пользователя хранятся в cookie, для работы с ними была использована библиотека jQuery Cookie с доработками (www/libs/jquery.cookie.js).

Вся работа таблицы расписания описывается в файле www/scripts/table\_script.js , а именно:

Прокручивание шапки влево-вправо, при прокручивании таблицы.

Прокручивание двух левых закрепленных столбцов вверх-вниз, при прокручивании таблицы.

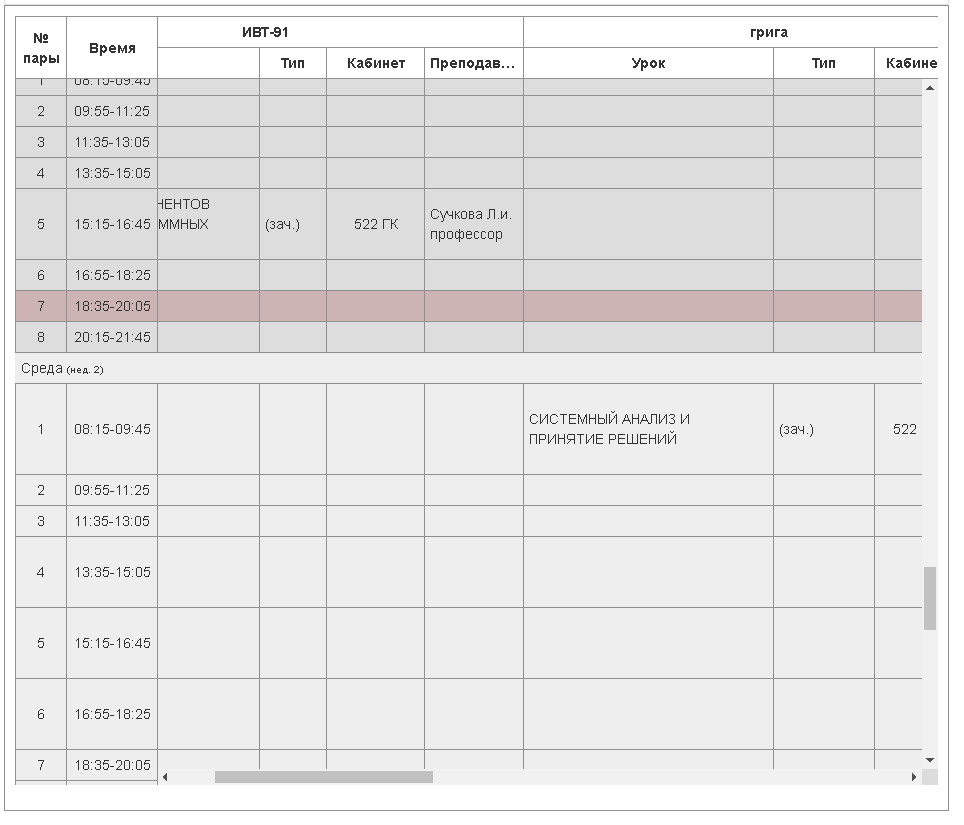


Рисунок 25 – Прокрутка закрепленных строк и столбцов

Возможность изменять размеры столбцов при помощи перетаскивании "палок", обозначающих границы столбцов.



Рисунок 26 – Изменение размера столбца

Установка размеров строк, данное решение является устранением ошибок отображения таблицы и закрепленных столбцов.

Перемещение к текущему дню или неделе при загрузки страницы.

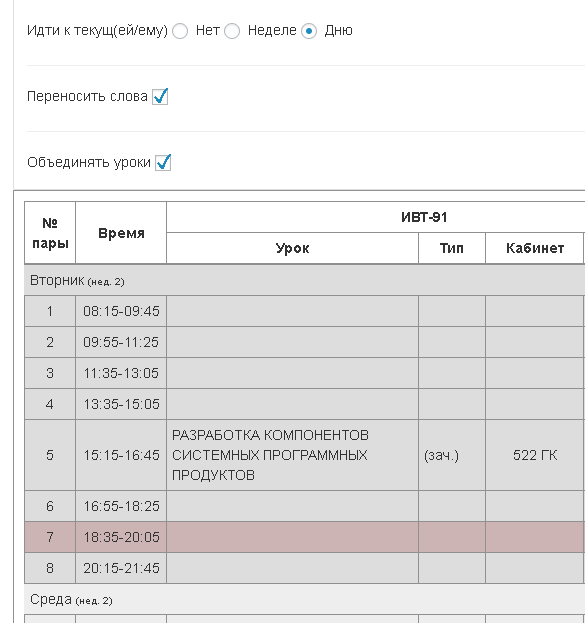


Рисунок 27 – Перемещение к текущему дню

Управление модальным окном для добавления новых разделов.

Управление сохранением настроек.

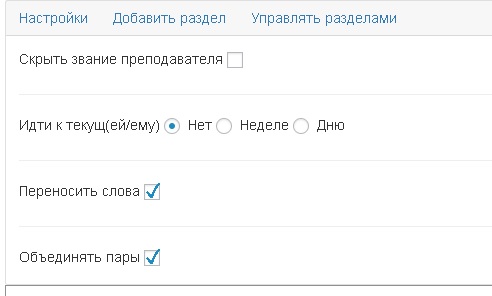


Рисунок 28 – Вкладка настроек

Управление настройками разделов (перемещение разделов, удаление разделов, перемещение столбцов, переключение видимости столбцов, установка названия раздела)

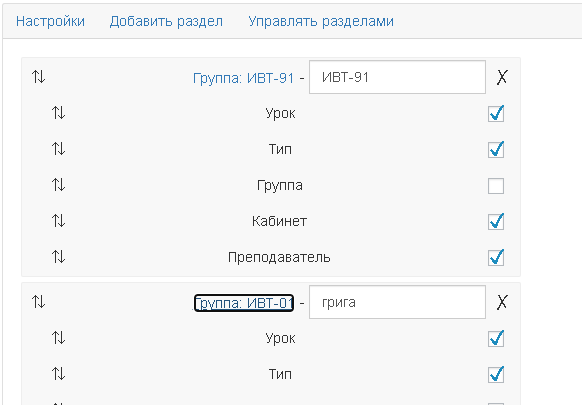


Рисунок 29 – Вкладка настроек разделов

Дополнительные технические скрипты.

На странице /login используются скрипты для регистрации, переключения окон входа/регистрации/восстановления пароля (www/scripts/login\_script.js), демонстрации пароля и отправки форм (www/scripts/script.js).

Скрипт регистрации представляет собой клиентскую проверку данных перед отправкой, проверку занятости введенного логина (с помощью отправки соответствующего ajax запроса на сервер), отправку данных на сервер с помощью ajax.

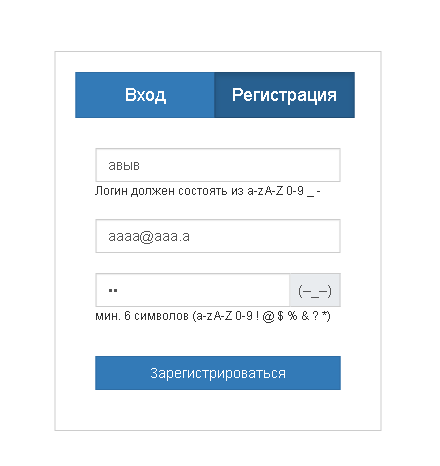


Рисунок 30 – Проверка введенных данных

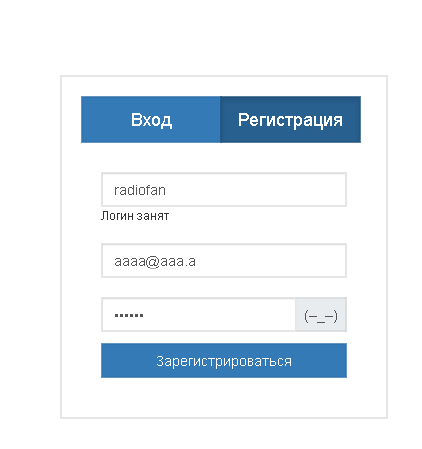


Рисунок 31 – Проверка занятости логина

Скрипт демонстрации пароля по нажатию на "лицо" меняет тип поля с пароля на текст и обратно.

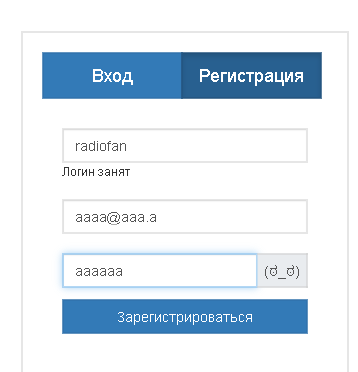


Рисунок 32 – Демонстрация пароля

Скрипт отправки форм при отправки формы собирает ее данные и отправляет их на сервер с помощью ajax запроса, не перезагружая страницу.

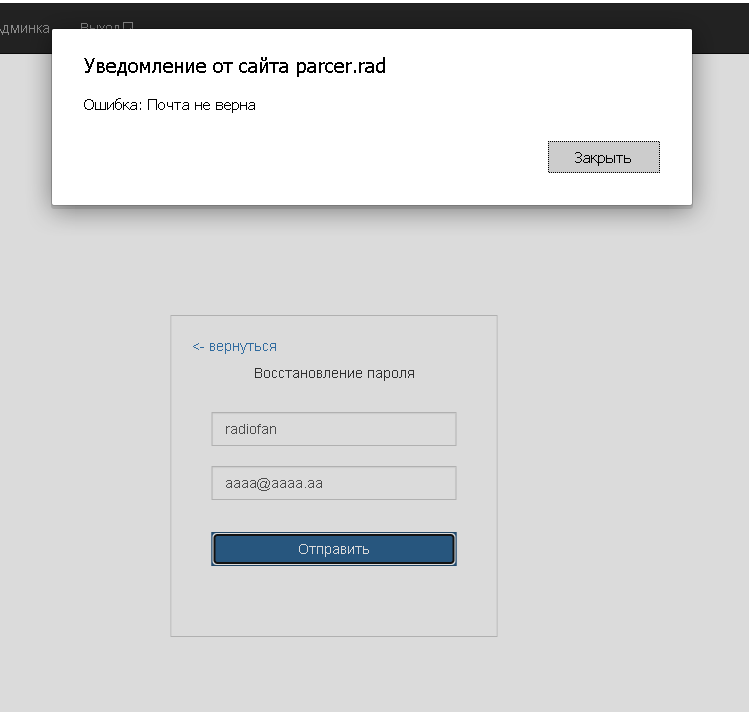


Рисунок 33 – Пример отправки формы с помощью ajax запроса и ответа сервера.

# Заключение

В процессе работы был создано web-приложение улучшающее работу с расписанием АлтГТУ:

* Спроектирована реляционная модель базы данных;
* База данных реализована на MySQL 5.7;
* Произведена верстка с помощью HTML и CSS;
* Собрано ядро приложения на PHP.

Достоинства разработанного WEB приложения:

* Удобное отображение данных;
* Актуальные данные;
* Гибкая настройка интерфейса;