

**Министерство образования и науки Российской Федерации**  
федеральное государственное автономное образовательное учреждение  
высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

---

Институт кибернетики  
Направление подготовки 09.03.01 Информатика и вычислительная техника  
Кафедра Информационные системы и технологии

**БАКАЛАВРСКАЯ РАБОТА**

Тема работы
<b>Система интерактивных информационных табличек аудиторий на базе платформы Raspberry Pi</b>

УДК 004.42:378.145

Студент

Группа	ФИО	Подпись	Дата
8В3А	Чайбар Артыш Алимович		

Руководитель

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент кафедры АиКС	Фадеев Александр Сергеевич	к.т.н.		

**КОНСУЛЬТАНТЫ:**

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент каф МЕН.	Антонова Ирина Сергеевна	к.э.н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Инженер	Маланова Наталья Викторовна	к.т.н.		

**ДОПУСТИТЬ К ЗАЩИТЕ:**

Зав. кафедрой	ФИО	Ученая степень, звание	Подпись	Дата
ИСТ	Мальчуков Андрей Николаевич	к.т.н.		

**ЗАПЛАНИРОВАННЫЕ РЕЗУЛЬТАТЫ ПО ОСНОВНОЙ  
ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЕ ПОДГОТОВКИ БАКАЛАВРОВ  
09.03.01 «ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА», ИК  
ТПУ, ПРОФИЛЬ «ВЫЧИСЛИТЕЛЬНЫЕ МАШИНЫ, КОМПЛЕКСЫ,  
СИСТЕМЫ И СЕТИ»**

Код результата тов	Результат обучения (выпускник должен быть готов)
<i>Профессиональные компетенции</i>	
P1	Применять базовые и специальные естественнонаучные и математические знания в области информатики и вычислительной техники, достаточные для комплексной инженерной деятельности.
P2	Применять базовые и специальные знания в области современных информационных технологий для решения инженерных задач.
P3	Ставить и решать задачи комплексного анализа, связанные с созданием аппаратно-программных средств информационных и автоматизированных систем, с использованием базовых и специальных знаний, современных аналитических методов и моделей.
P4	Разрабатывать программные и аппаратные средства (системы, устройства, блоки, программы, базы данных и т. п.) в соответствии с техническим заданием и с использованием средств автоматизации проектирования.
P5	Проводить теоретические и экспериментальные исследования, включающие поиск и изучение необходимой научно-технической информации, математическое моделирование, проведение эксперимента, анализ и интерпретация полученных данных, в области создания аппаратных и программных средств информационных и автоматизированных систем.
P6	Внедрять, эксплуатировать и обслуживать современные программно-аппаратные комплексы, обеспечивать их высокую эффективность, соблюдать правила охраны здоровья, безопасность труда, выполнять требования по защите окружающей среды.
<i>Универсальные компетенции</i>	
P7	Использовать базовые и специальные знания в области проектного менеджмента для ведения комплексной инженерной деятельности.
P8	Владеть иностранным языком на уровне, позволяющем работать в иноязычной среде, разрабатывать документацию, презентовать и защищать результаты комплексной инженерной деятельности.
P9	Эффективно работать индивидуально и в качестве члена группы, состоящей из специалистов различных направлений и квалификаций, демонстрировать ответственность за результаты работы и готовность следовать корпоративной культуре организации.
P10	Демонстрировать знания правовых, социальных, экономических и культурных аспектов комплексной инженерной деятельности.
P11	Демонстрировать способность к самостоятельному обучению в течение всей жизни и непрерывному самосовершенствованию в инженерной профессии.

**Министерство образования и науки Российской Федерации**  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

---

Институт Кибернетики  
Направление подготовки (специальность) 09.03.01 Информатика и вычислительная техника  
Кафедра Информационные системы и технологии

УТВЕРЖДАЮ:

Зав. кафедрой

\_\_\_\_\_  
(Подпись)

\_\_\_\_\_  
(Дата)

А.Н. Мальчуков  
(Ф.И.О.)

**ЗАДАНИЕ**

**на выполнение выпускной квалификационной работы**

В форме:

бакалаврской работы
---------------------

Студенту:

Группа	ФИО
8В3А	Чайбар Артышу Алимовичу

Тема работы:

<b>Система интерактивных информационных табличек аудиторий на базе платформы Raspberry Pi</b>	
Утверждена приказом директора (дата, номер)	От 07.02.2017 №790/с

Срок сдачи студентом выполненной работы:	19.06.2017
--	------------

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ:**

<b>Исходные данные к работе</b>	Создание клиент-серверного приложения, которое формирует информацию для определенной аудитории и выводит полученную информацию на дисплей в режиме реального времени.
<b>Перечень подлежащих исследованию, проектированию и разработке вопросов</b>	Проведение анализа современных программ для написания веб-приложения, программного обеспечения, позволяющее упростить разработку графического интерфейса приложения, анализ программ для отображения приложения, анализ и изучение работы СУБД MySQL и её

	взаимодействия с клиентом.
<b>Перечень графического материала</b>	Скриншоты интерфейса программ.
<b>Консультанты по разделам выпускной квалификационной работы</b>	
<b>Раздел</b>	<b>Консультант</b>
<b>Социальная ответственность</b>	Антонова Ирина Сергеевна
<b>Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</b>	Маланова Наталья Викторовна
<b>Названия разделов, которые должны быть написаны на русском и иностранном языках:</b>	
Всё разделы должны быть написаны на русском языке.	

<b>Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику</b>	10.11.2016
---	------------

**Задание выдал руководитель:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент кафедры АиКС	Фадеев Александр Сергеевич	к.т.н		

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8В3А	Чайбар Артыш Алимович		

**Министерство образования и науки Российской Федерации**  
 Федеральное государственное бюджетное образовательное учреждение  
 высшего профессионального образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
 ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Институт Кибернетики  
 Направление подготовки 09.03.01 Информатика и вычислительная техника  
 Уровень образования Бакалавриат  
 Кафедра Информационных систем и технологий  
 Период выполнения осенний / весенний семестр 2016/2017 учебного года

Форма представления работы:

Бакалаврская работа
---------------------

**КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН**  
**выполнения выпускной квалификационной работы**

Срок сдачи студентом выполненной работы:	19.06.2017
--	------------

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
10.11.2016	Анализ предметной области	15
15.12.2016	Проектирование	20
06.03.2017	Реализация	30
14.04.2017	Тестирование	15
9.06.2017	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	10
8.06.2017	Социальная ответственность	10

Составил преподаватель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент кафедры АиКС	Фадеев Александр Сергеевич	к.т.н.		

**СОГЛАСОВАНО:**

Зав. кафедрой	ФИО	Ученая степень, звание	Подпись	Дата
Информационных систем и технологий	Мальчуков Андрей Николаевич	к.т.н.		

## РЕФЕРАТ

Выпускная квалификационная работа выполнена на 70 страницы, содержит 34 рисунка, 16 таблиц, 12 источников, 5 приложений.

Ключевые слова: расписание, интерактивная табличка, клиент-серверное приложение.

Объектом работы является изучение способов отображения требуемой информации об аудиториях на интерактивных дисплеях.

Цель работы – создание клиент-серверного приложения, которое формирует информацию для определенной аудитории и выводит полученную информацию на дисплей в режиме реального времени.

В процессе разработки проводился анализ современных программ для написания веб-приложения, программного обеспечения, позволяющее упростить разработку графического интерфейса приложения, анализ программ для отображения приложения, анализ и изучение работы СУБД MySQL и её взаимодействия с клиентом.

В результате исследования были выбраны: программное обеспечение для реализации приложения Brackets, фреймворк Bootstrap для разработки графического интерфейса, приложение Google Chrome для просмотра, языки программирования PHP, JavaScript, интернет-технологии HTML, CSS.

Основные конструктивные, технологические и технико-эксплуатационные характеристики: поддержка языков PHP, MySQL, HTML, CSS, JavaScript.

Степень внедрения: опытная эксплуатация.

Область применения:

- в офисных помещениях,
- для оперативного информирования о статусе должностных лиц в кабинетах,
- в учебных заведениях для информации и занятии, проводимом в аудитории,

- в медицинских учреждениях, которые предъявляют повышенные требования в оперативной информации о том или ином кабинете для посетителей,
- в социальных и сервисных службах, где требуется упорядочивание посещения кабинетов посетителями и др.

В будущем планируется:

- расширение функционала отображаемой информации в автоматическом режиме (интеграция с личным электронным календарем должностного лица, базами данных университета, добавлением возможности вывода сопроводительной информации, например, об исторической ценности самой аудитории, об известных профессорах, работавших в ней и др.)
- расширение функционала управления отображаемой информацией, задаваемой в ручном режиме: интеграция с информационными системами предприятия, получение сведений о сотруднике, использование локальных систем аутентификации пользователей, интеграция с API социальных сетей (Вконтакте).
- создание гибких шаблонов легкочитаемого красочного (анимированного) оформления изображения интерактивных табличек; адаптация информационного содержимого к дисплеям различного размера и разрешения
- разработка линейки законченных устройств, включая цветной дисплей, встраиваемый микрокомпьютер, адаптеры электропитания, сетевые интерфейсы.

## **ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ, СОКРАЩЕНИЯ И НОРМАТИВНЫЕ ССЫЛКИ**

**Интернет-браузер** – программное обеспечение для просмотра веб-страниц.

**СУБД** – система управления базами данных.

**Тонкий клиент** – компьютер или программа-клиент в сетях с клиент-серверной или терминальной архитектурой, который переносит все или большую часть задач по обработке информации на сервер.

**Live Search** – это поиск, при котором результат запроса сразу отображается на странице, при вводе даже одной буквы. С вводом следующего символа, предлагается более точный результат запроса.

### **ОБЪЕКТ И МЕТОД ИССЛЕДОВАНИЯ**

Объектом исследования является информационная система для аудиторий на базе Raspberry Pi, а предметом исследования – совокупность элементов информационного обеспечения, связей и отношений между подсистемами, компонентами и элементами информационной системы.

Методом исследования является анализ информационной системы, моделирование базы данных для работы клиент-серверного приложения.



# ОГЛАВЛЕНИЕ

<b>ВВЕДЕНИЕ .....</b>	<b>12</b>
<b>1. Обзор интернет технологий.....</b>	<b>13</b>
1.1. Обзор Raspberry Pi .....	15
1.2. Языки программирования .....	17
1.2.1. Обзор JavaScript.....	18
1.2.2. Обзор PHP .....	19
1.2.3. CSS-фреймворки .....	20
1.3. Среда разработки .....	21
<b>2. Проектирование информационной системы .....</b>	<b>24</b>
2.1. Структура информационной системы .....	24
2.2. Проектирование базы данных для статуса кабинетов/аудиторий .....	24
2.2.1. Таблица rasp .....	26
2.2.2. Таблица users .....	26
2.2.3. Таблица groups.....	26
2.2.4. Таблица class.....	26
2.2.5. Таблица lessons .....	27
2.2.6. Таблица tim_lesson .....	27
2.2.7. Таблица week .....	27
2.3. Проектирование базы данных для статуса должностных лиц .....	29
2.3.1. Таблица users .....	29
2.3.2. Таблица class.....	30
2.4. Проектирование структуры дисплея.....	30
2.4.1. Страница учебной аудитории .....	31
2.4.2. Страница для преподавательских аудиторий.....	33
2.5. Проектирование клиентской части .....	34
<b>3. Реализация информационной системы .....</b>	<b>36</b>
3.1. Разработка системы для статуса кабинета/аудитории .....	36
3.2. Разработка системы для статуса должностного лица .....	40
3.3. Реализация панели управления администратора.....	44
3.3.1. Профиль пользователя: редактирование и удаление.....	45
3.3.2. Добавить пользователя .....	48

3.3.3.	Поиск пользователей в системе .....	50
3.4.	Настройка и работа системы на платформе Raspberry PI .....	51
<b>4.</b>	<b>Социальная ответственность .....</b>	<b>56</b>
4.1.	Введение .....	57
4.2.	производственная безопасность .....	58
4.2.1.	Освещенность рабочей зоны .....	59
4.2.2.	Производственный шум .....	61
4.2.3.	Микроклимат помещения.....	63
4.2.4.	Электромагнитное излучение .....	64
4.2.5.	Поражение электрическим током.....	65
4.3.	Пожарная безопасность.....	66
4.4.	Экологическая безопасность .....	67
4.5.	Безопасность в чрезвычайных ситуациях .....	67
4.6.	Правовые и организационные вопросы обеспечения безопасности.....	69
<b>5.</b>	<b>Финансовый менеджмент, ресурсоэффективность и ресурсосбережение.....</b>	<b>74</b>
5.1.	Введение .....	74
5.2.	Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения .....	75
5.2.1.	Технология QuaD .....	75
5.2.2.	SWOT-анализ.....	76
5.3.	Определение возможных альтернатив проведения научных исследований.....	77
5.4.	Планирование научно-исследовательских работ .....	77
5.4.1.	Структура работ в рамках научного исследования .....	77
5.4.2.	Определение трудоемкости выполнения работ .....	78
5.4.3.	Разработка графика проведения научного исследования .....	79
5.5.	Бюджет научно-технического исследования .....	82
5.5.1.	Расчет материальных затрат НИИ .....	82
5.5.2.	Расчет затрат на специальное оборудование для научных (экспериментальных) работ	84
5.5.3.	Основная заработная плата исполнителей темы.....	84
5.5.4.	Отчисления во внебюджетные фонды .....	86
5.5.5.	Расчет затрат на научные и производственные командировки .....	86
5.5.6.	Контрагентные расходы .....	87

5.5.7. Накладные расходы.....	87
5.5.8. Формирование бюджета затрат научно-исследовательского проекта.....	87
<b>Заключение .....</b>	<b>89</b>
<b>Список публикаций автора .....</b>	<b>90</b>
<b>Список использованных источников .....</b>	<b>91</b>
<b>Приложение А. Листинг клиент-серверного приложения для статуса кабинета/аудитории .....</b>	<b>94</b>
<b>Приложение Б. Листинг клиент-серверного приложения для статуса должностного лица .....</b>	<b>99</b>
<b>Приложение В. Листинг панели управления администратора.....</b>	<b>104</b>
<b>Приложение Г. Листинг скрипта создания пользователя.....</b>	<b>110</b>
<b>Приложение Д. Поиск пользователей в системе .....</b>	<b>111</b>

## **ВВЕДЕНИЕ**

Сегодня расписание занятий студентов ТПУ предоставлено в виде веб-списка, доступ к которому возможен с помощью мобильного приложения «расписание вузов», с мобильного или стационарного компьютера, или электронного терминала, установленного в корпусах университета.

Несмотря на то, что интерактивная система расписаний занятий ТПУ имеет богатый функционал, она располагает некоторыми недостатками. Для проверки текущего расписания необходимо запустить программу-браузер или мобильное приложение и иметь подключение к Интернету.

Дверные таблички, используемые на дверях каждой аудитории, требуют расходов ресурсов и дополнительного времени на их изготовление. В большинстве случаев, изготовление дверных табличек или оформление табличек – это процесс, требующий достаточно времени, что является не приемлемым для работника.

Основной задачей данной работы является повысить оперативную информативность дверных табличек, исключить необходимость поиска расписания студентом в момент начала занятий в определенной аудитории и исключить процесс изготовления новых дверных табличек для кабинетов сотрудников при смене должностных лиц. Предполагаемое решение поставленной задачи – отображение текущего расписания и информации о кабинете сотрудника на интерактивном дисплее, который должен быть установлен возле каждой аудитории.

Целью работы является создание информационной системы для учебных аудиторий и кабинетов сотрудников, которая в интерактивном режиме будет отображать оперативную информацию о сотруднике и аудитории.

# 1. ОБЗОР ИНТЕРНЕТ ТЕХНОЛОГИЙ

Задача данной работы была разработка информационной системы, которая должна выводить информацию для аудиторий и о статусе должностного лица.

Система для аудиторий должна выводить информацию о расписании для конкретной аудитории, около которой установлена. В перечень выводимой информации входят следующие значения:

1. Номер аудитории;
2. ФИО преподавателя;
3. Тип занятия;
4. Номер группы/групп

Данная информация должна отображаться в блоках, которые отвечают за порядок занятий в аудитории, первый блок отображает информацию предыдущего занятия, второй – текущее занятие, третий – следующее занятие. Выходные данные формируются клиент-серверным приложением, которые берутся с сервера университета через SQL запросы к БД университета.

Система статуса должностных лиц должна обеспечивать оперативное информирование: номер, название кафедры, ФИО, должность, статус. Статус отображает сотрудника: отсутствие или присутствие на рабочем месте.

Управление выводом информации на дисплей должно обеспечиваться функционалом личного кабинета. Для входа в личный кабинет требуются персональные логин и пароль сотрудника.

Вся информация должна выводиться на дисплей, который подключен к Raspberry PI. Он является тонким клиентом, который подключен к сети интернет. Схема запроса на сервер от тонкого клиента на получение данных показана на рисунке 1:

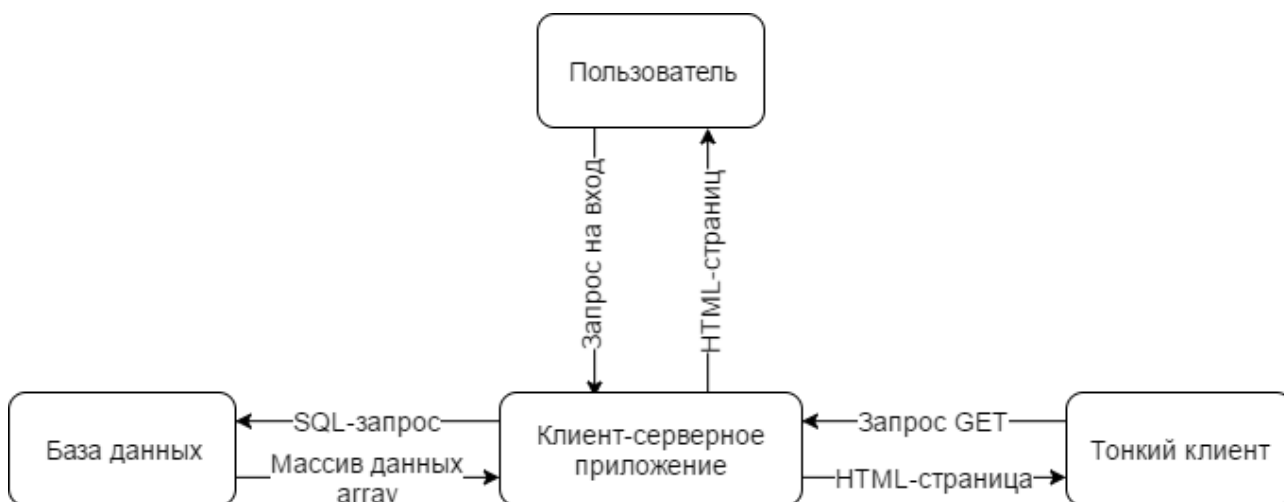


Рисунок 1. Схема получения данных

Как показано на рисунке 1, тонкий клиент отправляет запрос GET к приложению. Запрос GET – это запрос, который открыто передается серверу через адресную строку браузера. Получив запрос, приложение формирует SQL запрос, который передается на сервер, где хранится база данных. В ответ с сервера получаем массив данных, которые обрабатываются клиент-серверным приложением.

После обработки полученных данных, формируется форма HTML-страницы, в которую передаются соответствующие данные, после чего выводится на дисплей.

В свою очередь, на тонкий клиент могут передаваться две формы: страница с расписание для аудитории и страница со статусом должностного лица. Структура тонкого клиента показана на рисунке 2.



Рисунок 2. Структура тонкого клиента

При авторизации пользователя на странице входа, формируется запрос на вход в личный кабинет пользователя, то есть запрос на проверку входных данных, таких как логин и пароль. Если после отправки запроса приложением на сервер, входные данные совпадают с данными из БД, отправляется массив данных пользователя, которые в свою очередь обрабатываются приложением и так же формируется форма HTML-страница, которая выводится на дисплей.

Возможны два параметра входа: вход в личный кабинет и вход на страницу администратора (рисунок 3).

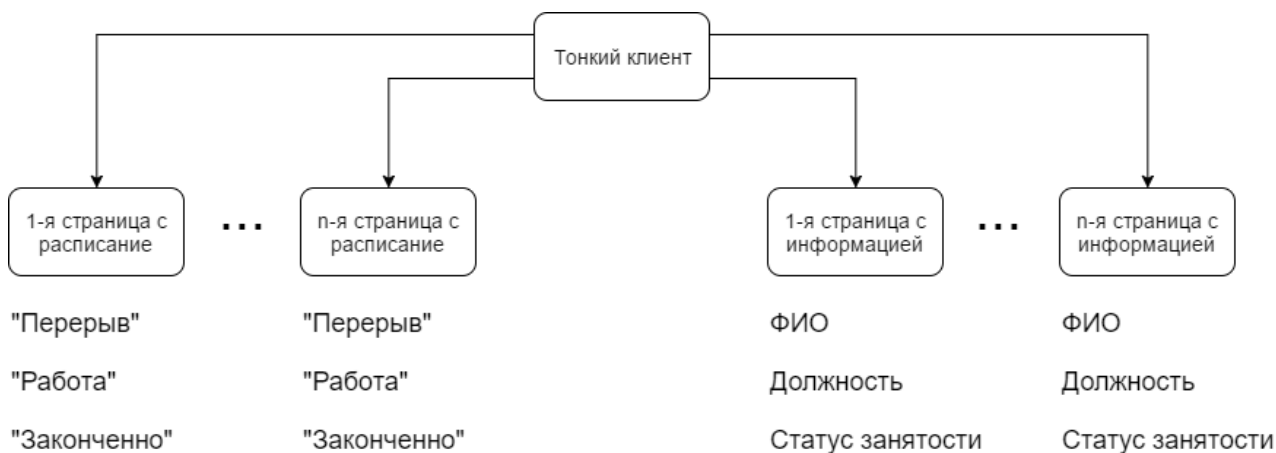


Рисунок 3. Структура пользователя

Как было ранее описано, личный кабинет отвечает за вывод информации на дисплей.

При авторизации как администратор, открывается доступ к следующим возможностям: просмотр списка пользователей системы, просмотр персональной информации пользователя и возможность обновление ее обновления, удаление пользователей, добавить пользователя, поиск и просмотр любой аудитории.

### 1.1. Обзор Raspberry Pi

Raspberry Pi 2 Model B – это миниатюрный, размером с кредитную карту, дешёвый компьютер.

В совокупности с низкими требованиями открытого программного обеспечения (OpenSource) к аппаратной части и специально собранным ядром операционной системы, оптимизированным под данное железо, это позволяет установить на него операционную систему Linux (либо RiscOS), а также набор сопутствующего программного обеспечения.

В сравнении с Raspberry Pi 1, второе поколение располагает:

- Процессор: четырехядерный ARM Cortex-A7 900MHz;
- Память: 1GB RAM.

Как и первое поколение располагает:

- 4 порта USB 2.0;
- 40 GPIO пинов;
- Полноценный HDMI;
- Интернет порт;
- Комбинированный 3,5-мм аудиоразъем и композитный видеосигнал;
- Интерфейс для подключения камеры (CSI);
- Интерфейс для подключения дисплея (DSI);
- Разъем для MicroSD;
- Графическое ядро VideoCore IV 3D.





Рисунок 4. Raspberry Pi 2 Model B

## 1.2. Языки программирования

Реализация данного проекта возможна на разных языках веб-программирования. На данный момент есть ряд самых используемых языков:

1. JavaScript
2. Java
3. PHP
4. Python
5. C#,
6. C++
7. Ruby
8. CSS
9. C

Для работы с клиентской стороны используются такие языки как JavaScript, CSS, C# и т.д. В данном случае были выбраны языки JavaScript,

CSS так как они больше ориентированы для создания интерактивности веб-страницам, на которых будет выводиться вся необходимая информация. Также являются самыми популярными. Создание самого интерфейса было основано на языке HTML с использованием фреймворка Bootstrap.

Для работы с сервером выбран язык PHP, так как располагает достаточным набором встроенных средств, и является популярным серверным языком программирования.

### **1.2.1. Обзор JavaScript**

JavaScript – это язык программирования, с помощью которого веб-страницам придается интерактивность. С его помощью создаются приложения, которые включаются в HTML-код (например, анкеты или формы регистрации, которые заполняются пользователем).

В браузере JavaScript умеет делать всё, что относится к манипуляции со страницей, взаимодействию с посетителем и, в какой-то мере, с сервером:

1. Создавать новые HTML-теги, удалять существующие, менять стили элементов, прятать, показывать элементы и т.п.
2. Реагировать на действия посетителя, обрабатывать клики мыши, перемещения курсора, нажатия на клавиатуру и т.п.
3. Посылать запросы на сервер и загружать данные без перезагрузки страницы (эта технология называется "AJAX").
4. Получать и устанавливать cookie, запрашивать данные, выводить сообщения

Несмотря на перечисленные возможности, браузер накладывает на его исполнение некоторые ограничения.

Это сделано для безопасности пользователей, чтобы злоумышленник не мог с помощью JavaScript получить личные данные или как-то навредить компьютеру пользователя.

1. JavaScript не может читать/записывать произвольные файлы на жесткий диск, копировать их или вызывать программы. Он не имеет прямого доступа к операционной системе.
2. JavaScript, работающий в одной вкладке, не может общаться с другими вкладками и окнами, за исключением случая, когда он сам открыл это окно или несколько вкладок из одного источника (одинаковый домен, порт, протокол).
3. Из JavaScript можно легко посылать запросы на сервер, с которого пришла страница. Запрос на другой домен тоже возможен, но менее удобен, т. к. и здесь есть ограничения безопасности.

Есть как минимум три замечательных особенности JavaScript:

1. Полная интеграция с HTML/CSS.
2. Простые вещи делаются просто.
3. Поддерживается всеми распространёнными браузерами и включён по умолчанию.

Этих трёх вещей одновременно нет больше ни в одной браузерной технологии.

Поэтому JavaScript и является самым распространённым средством создания браузерных интерфейсов.

### **1.2.2. Обзор PHP**

PHP (Hypertext Preprocessor) – это серверный язык программирования общего назначения с открытым исходным кодом. PHP сконструирован специально для ведения Web-разработок и его код может внедряться непосредственно в HTML.

Вместо рутинного вывода HTML-кода командами языка, скрипт PHP содержит HTML с вкраплениями кода. Код PHP отделяется специальными начальным и конечным тегами, которые позволяют "переключаться" в "PHP-режим" и выходить из него.

PHP отличается от JavaScript тем, что PHP-скрипты выполняются на сервере и генерируют HTML, который посылается клиенту. Если бы на сервере был размещен скрипт, клиент получил бы только результат его выполнения, но не смог бы выяснить, какой именно код его произвел. Возможно настроить сервера таким образом, чтобы HTML-файлы обрабатывались процессором PHP, в результате получим web-страницу, которая никак не будет отличаться классической HTML-страницы.

PHP прост для освоения и способен удовлетворить запросы профессиональных программистов.

### 1.2.3. CSS-фреймворки

**Фреймворк** – это программная оболочка, позволяющая упростить и ускорить решение типовых задач, характерных для данного языка программирования [2].

**Bootstrap** (также известен как Twitter Bootstrap) – свободный набор инструментов для создания сайтов и веб-приложений. Включает в себя HTML- и CSS-шаблоны оформления для типографики, веб-форм, кнопок, меток, блоков навигации и прочих компонентов веб-интерфейса, включая JavaScript-расширения [3].

Преимущества bootstrap:

1. Скорость работы – благодаря множеству готовых элементов вёрстка с bootstrap занимает значительно меньше времени;
2. Масштабируемость – добавление новых элементов не нарушает общую структуру;
3. Легко настраивается – редактирование стилей производится путём создания новых CSS-правил, которые исполняются вместо стандартных. При этом не требуется использовать атрибуты типа “!important”;
4. Большое количество шаблонов;
5. Огромное сообщество разработчиков;

6. Широкая сфера применения – Bootstrap используется в создании тем для практически любой CMS (OpenCart, Prestashop, Magento, Joomla, Bitrix, WordPress и любые другие), в том числе для одностраничных приложений.

### **1.3. Среда разработки**

На сегодняшний день есть много программ для редактирования кода и написания веб-приложений. При выборе оценивались ряд характеристик:

1. Удобство использования;
2. Мощность;
3. Расширяемость;
4. Эстетика;
5. Скорость;
6. Кросс-платформенность;
7. Цена.

Следует выделить несколько программ, подходящих под данные характеристики: NotePad++, Brackets, Netbeans.

NotePad++ это мощный и легкий редактор, который пользуется Windows. Он прост для новичков, но он также подходит для профессионалов. Notepad++ поддерживает каждый из популярных языков программирования. Поддерживает разбиение рабочей среды на два окна редактирования, FTP браузер, макросы и мощные возможности редактирования текста. Notepad++ абсолютно бесплатный (рисунок 1).

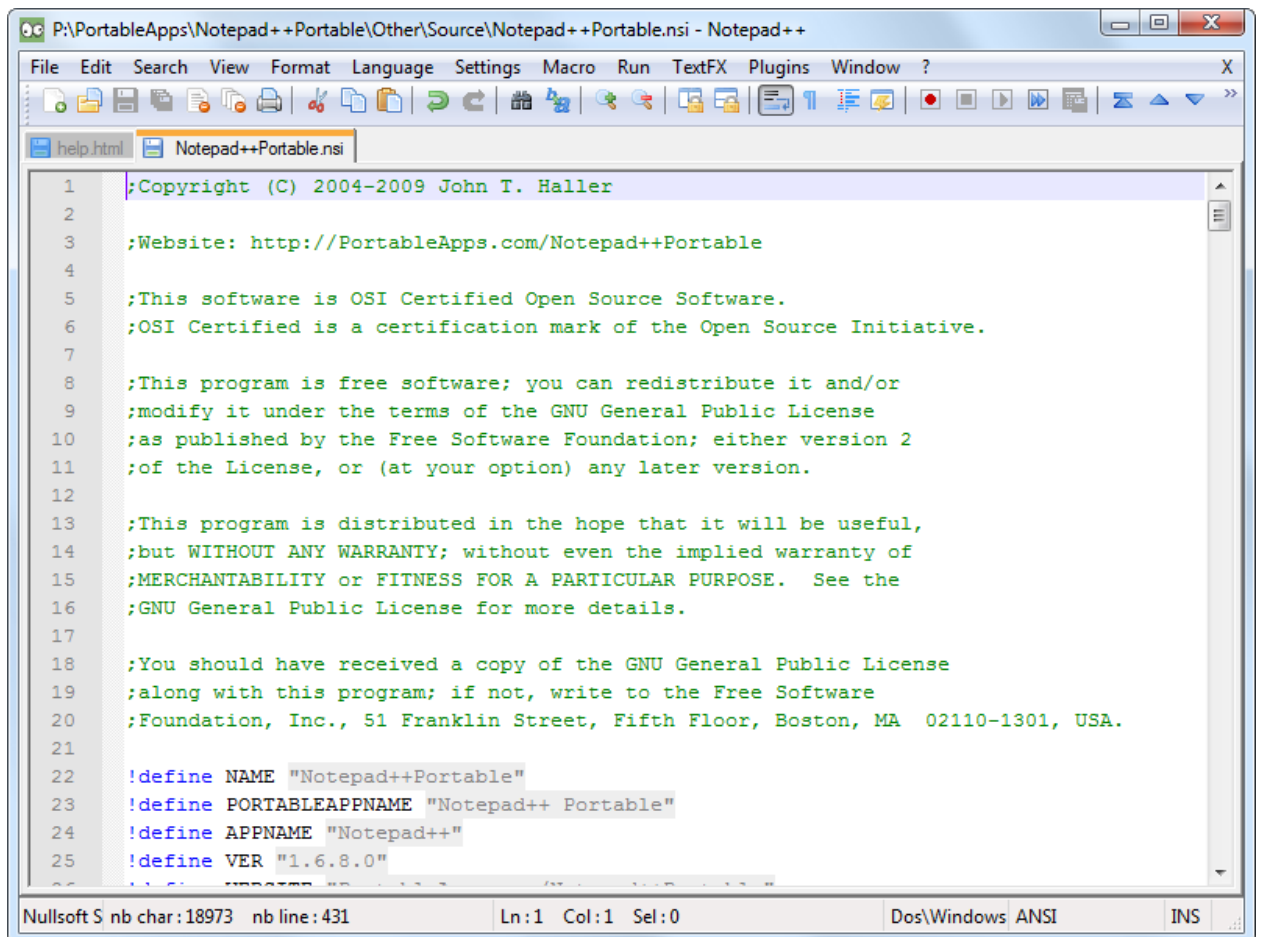


Рисунок 5. Окно программы Notepad++

Brackets является инструментом для веб-разработчиков и веб-дизайнеров. Приложение представляет собой легкий, удобный в использовании, но мощный, современный редактор кода, поддерживает подключение расширений и плагинов, которые позволяют упростить и ускорить работу при написании кода для веб-приложений. Brackets является бесплатным (рисунок 2).

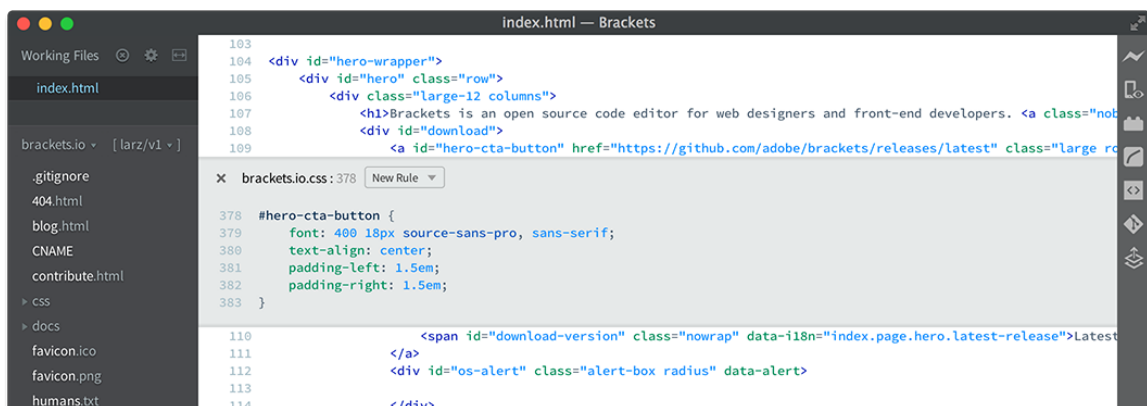


Рисунок 6. Окно программы Brackets

Netbeans среда для Java. Может быть расширена для дополнительных языков: PHP, Python, C/C++ и других. Приложение запускается на Linux, Windows и OSX. Netbeans может ускорить процесс разработки десктопных приложений с помощью Drag and Drop конструктора. Негативной стороной является производительность – Netbeans может тормозить на больших проектах. Этот редактор кода бесплатный и открытый ресурс (рисунок 3).

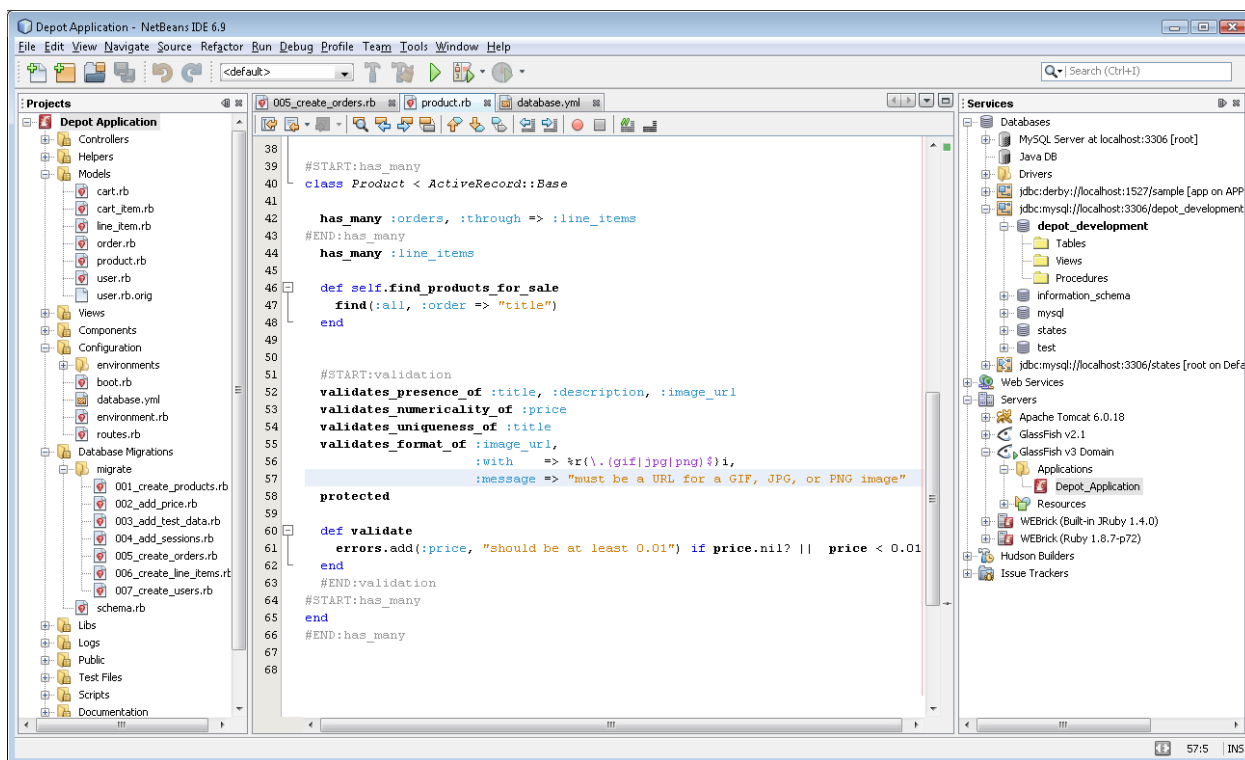


Рисунок 7. Окно программы Netbeans

Исходя из этого, была выбрана программа Brackets, так как удовлетворяет всем требованиям для разработки веб-приложения. Не исключено, что другие приложения могут располагать более мощной платформой, но для полноценной работы с ними надо платить [4].

## **2. ПРОЕКТИРОВАНИЕ ИНФОРМАЦИОННОЙ СИСТЕМЫ**

### **2.1. Структура информационной системы**

Система интерактивных информационных дверных табличек представляет собой распределенную информационную систему клиент-серверной архитектуры, состоящую из одного сервера и множества тонких клиентов на базе встраиваемых компьютеров Raspberry Pi и миниатюрных цветных дисплеев. Каждая интерактивная табличка размещается возле (или на) двери кабинета и в режиме реального времени отображает статус должностного лица либо состояние кабинета/аудитории. Управление информацией каждой таблички осуществляется через центральный веб-сервер, при этом имеется возможность, как формирования отображаемой информации в ручном режиме, так и в автоматическом, путем обработки данных внешних серверов и информационных систем.

Актуальность проекта обусловлена отсутствием аналогов, как для учебных аудиторий, так для кабинетов должностных лиц, низкой ценой отдельных устройств, легкостью масштабирования системы, гибкостью и адаптивностью.

### **2.2. Проектирование базы данных для статуса кабинетов/аудиторий**

При проектировании базы данных требовалось определить, какие данные надо хранить и как организовать связь между ними. В результате анализа, были перечислены все объекты базы данных и было решено использовать для системы отображения состояния кабинета/аудитории структуру, представленная на рисунке 8:

- Расписание (rasp);
- Пользователи (users);
- Группы (groups);
- Аудитория (class);
- Занятия (lessons);



- Время занятий (time\_lesson);
- Неделя (week).

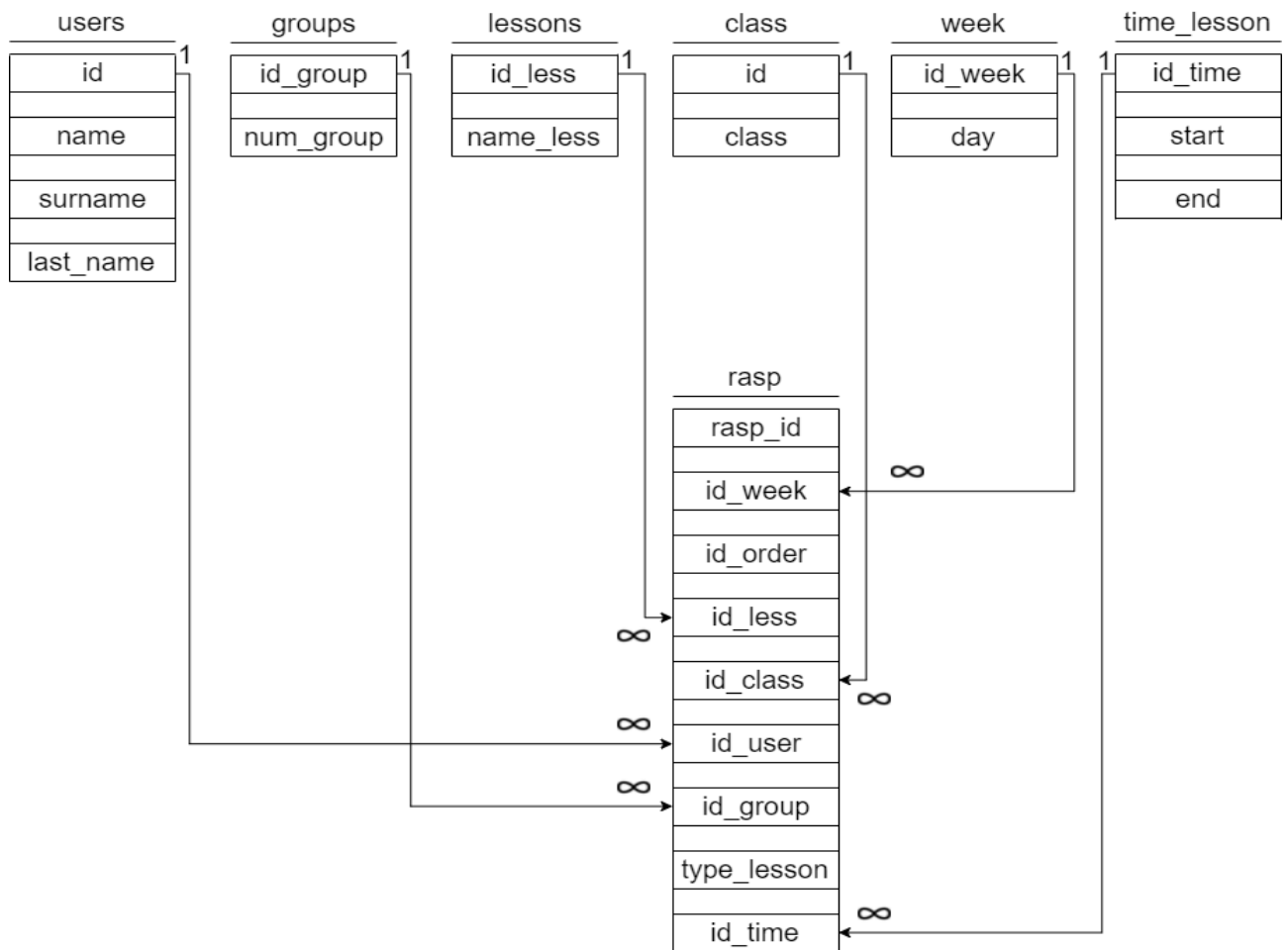


Рисунок 8. Логическая структура

Как видно из рисунка 8, ключевая таблица является rasp, так она связана со всеми элементами, которые требуются для формирования расписания, и хранит в себе порядок занятий для конкретной аудитории/кабинета.

Стоит отметить, что между таблицами использовалась связь один-многим (1:M). Это значит, что у таблицы (rasp) может быть несколько повторяющихся записей других таблиц, но каждой записи остальных таблиц, может соответствовать только одна запись таблицы (rasp). Эта связь будет удобна в случае, когда в одной аудитории проходит занятия для нескольких групп, значит, требуется использовать несколько одинаковых записей из таблиц остальных таблиц, например (users).

Представленная логическая структура состоит из 7 таблиц, которые описаны ниже с примерами записей, которые использовались в данной работе.

### 2.2.1. Таблица rasp

Данная таблица хранит в себе номера id таблиц, что в совокупности формируют запись для расписания, а добавление большего количества записей сформируют кортеж, что будет являться расписанием. Обращаясь по идентификаторам, получаем полные данные об атрибутах данной таблицы.

Таблица 1. Расписание (rasp)

rasp_id	id_week	id_order	id_less	id_class	id_user	id_group	type_lesson	id_time
Первичный ключ								
1	2	2	14	8	8	3	1	2

### 2.2.2. Таблица users

Таблица users отвечает за хранение данных о преподавателях: имя, фамилия и отчество.

Таблица 2. Пользователи (users)

Id	Name	Surname	Last_name
Первичный ключ			
1	Иван	Сидоров	Александрович

### 2.2.3. Таблица groups

Данная таблица отвечает за хранение номеров учебных групп.

Таблица 3. Группы (groups)

Id_group	Num_group
Первичный ключ	
1	8B51

### 2.2.4. Таблица class

Здесь хранятся данные об учебных аудиториях/кабинетах, а именно их номеров.

Таблица 4. Аудитория (class)

Id	Class
Первичный ключ	
1	307

### 2.2.5. Таблица lessons

Таблица lessons хранит перечень учебных дисциплин.

Таблица 5. Занятия (lessons)

Id_less	Name_less
Первичный ключ	
1	Математический анализ

### 2.2.6. Таблица tim\_lesson

В данной таблице расписано расписание звонков, что позволило системе определять начало и конец занятия.

Таблица 6. Время занятий (time\_lesson)

Id_time	Start	End
Первичный ключ		
1	8.30	10.05

### 2.2.7. Таблица week

Здесь хранятся перечень дней недели, чтобы отслеживать, какое расписание требуется выводить в определенный день.

Таблица 7. Неделя (week)

Id_week	Day
Первичный ключ	
1	Понедельник

После получения информационных объектов, требовалось определить физическую структуру база данных.

При ее составлении использовались типы varchar и int.

INT – это 4-х битный тип данных, который может принимать как отрицательные и положительные значения и имеет диапазон от -2147483648 до 2147483648. Был выбран данный тип, так как в основном он используется

во всех MySQL, и сравнения его с типами MEDIUMINT и BIGINT, INT является самым оптимальным, так как MEDIUMINT имеет низкий предел по максимальному размеру, а BIGINT слишком велик и требует 1 байт памяти.

VARCHAR – тип данных, который хранит строки переменной длины. При хранении величин типа VARCHAR используется только то количество символов, которое необходимо, плюс один байт для записи длины. Хранимые величины пробелами не дополняются, наоборот, концевые пробелы при хранении удаляются.

В результате анализа, получилась следующая физическая структура.

Таблица 8. Расписание (rasp)

rasp_id	id_week	id_order	id_less	id_class	id_user	id_group	Type_lesson	id_time
INT(11)	INT(11)	INT(11)	INT(11)	INT(11)	INT(11)	INT(11)	VARCHAR(30)	INT(11)

Таблица 9. Пользователи (users)

Id	Name	Surname	Last_name
INT(11)	VARCHAR(10)	VARCHAR(10)	VARCHAR(15)

Таблица 10. Группы (groups)

Id_group	Num_group
INT(11)	VARCHAR(10)

Таблица 11. Аудитория (class)

Id	Class
INT(11)	VARCHAR(5)

Таблица 12. Занятия (lessons)

Id_less	Name_less
INT	VARCHAR(255)

Таблица 13. Время занятий (time\_lesson)

Id_time	Start	End
INT(11)	VARCHAR(255)	VARCHAR(255)

Таблица 14. Неделя (week)

Id_week	Day
INT(11)	VARCHAR(20)

### 2.3. Проектирование базы данных для статуса должностных лиц

В ходе анализа данной системы, было решено, что в базе должно храниться полная информация о должностном лице, так как полное информирование одна из основных задач данной системы. Данная информация включает в себя полное ФИО, должность, рабочее место, то есть номер учебного корпуса и аудитории.

Учитывая, что каждый пользователь должен авторизоваться для работы в системе, так же надо было хранить данные о логине и пароле пользователя.

Так же требовалась информация о каждой аудитории. Это позволило выводить дополнительную информацию, например, тип или ее название. Были определены следующие атрибуты, описывающие любой кабинет: номер учебного корпуса, номер аудитории, информация по данной аудитории, тип аудитории.

Исходя из вышеперечисленных аспектов, которые были нужны для данной системы, были использованы выше описанные таблицы users и class. Надо было дополнить недостающими атрибутами, которые были описаны выше. После редактирования таблиц, получилась следующая структура:

#### 2.3.1. Таблица users

Таблица 15. Пользователи (users)

id	INT(11) PRIMARY KEY
name	VARCHAR(10)
surname	VARCHAR(10)
last_name	VARCHAR(15)
login	VARCHAR(30)
pass	VARCHAR(50)
class	VARCHAR(5)

build	VARCHAR(3)
info	VARCHAR(200)
active	VARCHAR(1)

### 2.3.2. Таблица class

Данная таблица хранит записи о каждом кабинете/аудитории.

Таблица 16. Аудитория (class)

id	class	build	name_class	type
INT(11)	VARCHAR(5)	VARCHAR(255)	VARCHAR(255)	VARCHAR(1)

Связь между таблицами используется один-ко-многим (1:M). Это значит, каждому кабинету/аудитории, может быть присвоено несколько сотрудников, но каждому сотруднику может принадлежать только один кабинет/аудитория. Логическая структура для данных таблиц показана на рисунке 9.

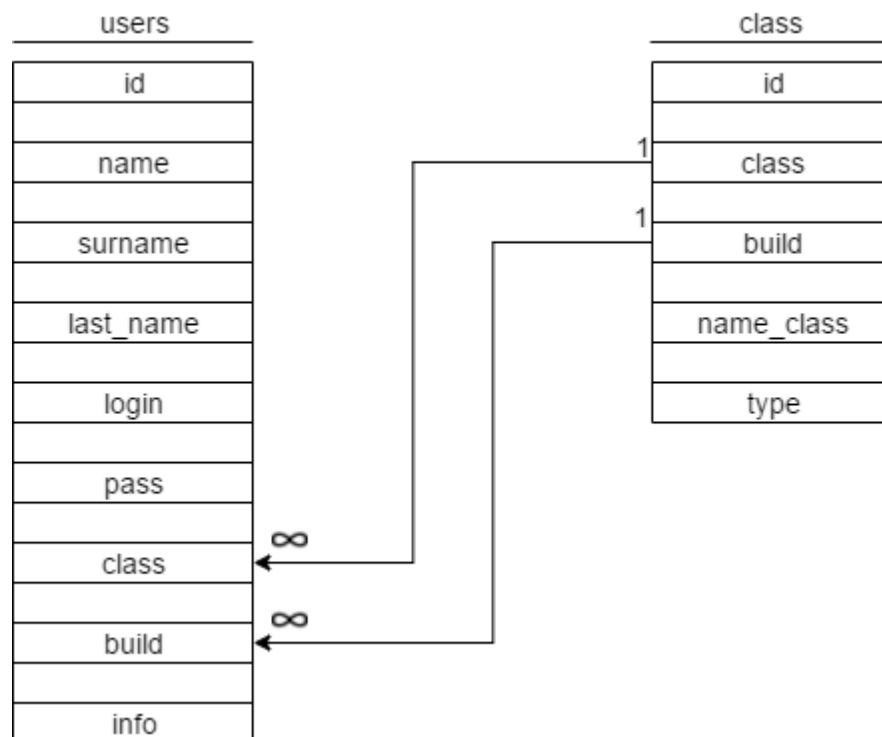


Рисунок 9. Логическая структура

## 2.4. Проектирование структуры дисплея

После окончания работы по созданию БД, требовалось разработать структуры страниц, которые выводились на дисплей. Они должны были

отвечать следующим требованиям: компактность, лаконичности, текст должен быть легко читаемым.

### **2.4.1. Страница учебной аудитории**

На рисунке 9 показан разработанный макет страницы, генерируемый клиент-серверным приложением после получение ответа с БД.

Каждый блок должен отображать нужную информацию в конкретный момент времени. Каждому пользователю будет отображаться номер аудитории, название дисциплины, номера групп, для которых проводится занятие и ФИО преподавателя. Соответствующая информация отображается в блоках следующего и предыдущего занятия.

Требовалось предусмотреть вывод информации в различные ситуации, названные состояние «перерыв» и «конец занятий» (рисунок 11, 12).

Если дисплей находится в состоянии «перерыв», то в блок текущего занятия выводится сообщение «Перерыв», в блок предыдущего занятия выводится только что прошедшее занятие, блок следующего занятия все так же выводит значение следующего занятия.

В состоянии «конец занятий» в блок текущего занятия должно выводиться сообщение «конец занятий», оставшиеся блоки остаются пустыми.

Рисунок 10. Структура дисплея для учебных аудиторий	Рисунок 11. Структура при состоянии «перерыв»	Рисунок 12. Структура дисплея при состоянии «конец занятий»
---	---	---

<p>Предыдущее занятие</p> <p><b>Аудитория</b></p> <p>Дисциплина Группы Преподаватель</p>	<p>Предыдущее занятие</p> <p><b>Аудитория</b></p> <p>Дисциплина Группы Преподаватель</p>	<p>Предыдущее занятие</p> <p><b>Аудитория</b></p> <p>Дисциплина Группы Преподаватель</p>
<p>Текущее занятие</p> <p><b>Аудитория</b></p> <p>Дисциплина Группы Преподаватель</p>	<p>Текущее занятие</p> <p><b>Аудитория</b></p> <p><b>Перерыв</b></p>	<p>Текущее занятие</p> <p><b>Аудитория</b></p> <p><b>Конец занятий</b></p>
<p>Предыдущее занятие</p> <p><b>Аудитория</b></p> <p>Дисциплина Группы Преподаватель</p>	<p>Предыдущее занятие</p> <p><b>Аудитория</b></p> <p>Дисциплина Группы Преподаватель</p>	<p>Предыдущее занятие</p> <p><b>Аудитория</b></p> <p>Дисциплина Группы Преподаватель</p>



## 2.4.2. Страница для преподавательских аудиторий

На рисунке 13 показан макет страницы для преподавательских аудиторий.

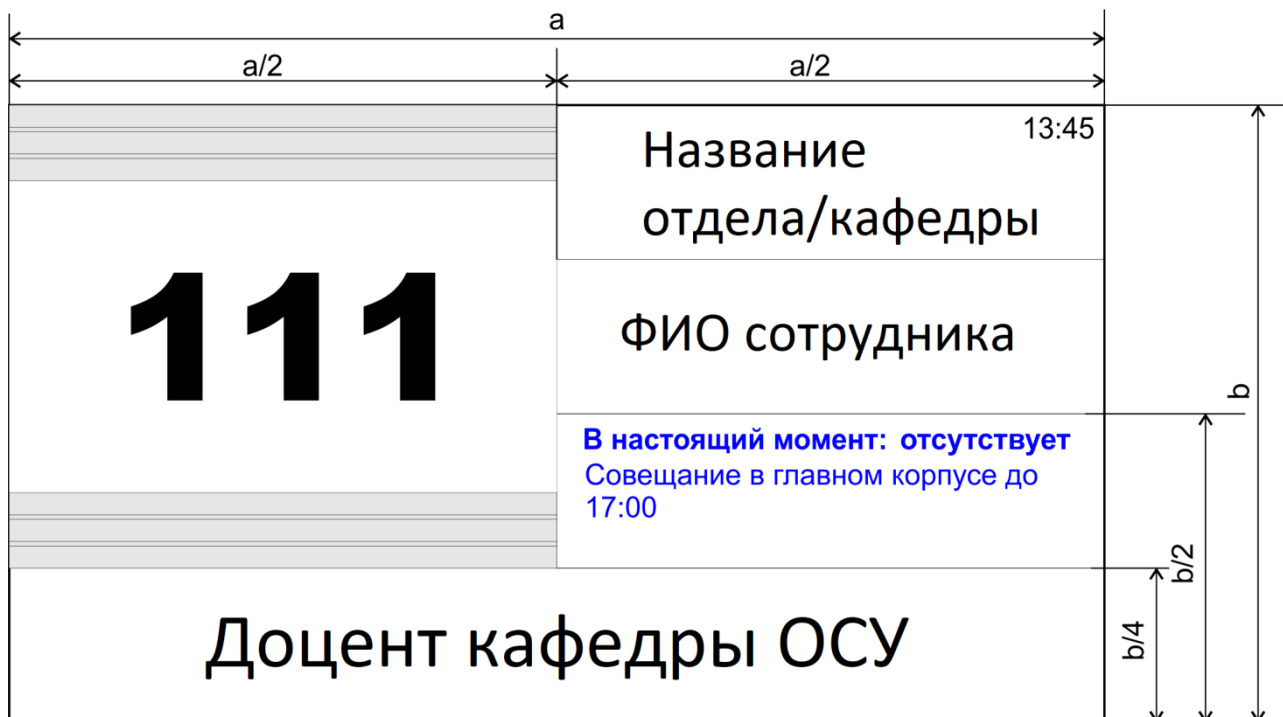


Рисунок 13. Макет для преподавательских аудиторий

Была предусмотрена работа дисплея в двух режимах:

- Сотрудник на месте;
- Сотрудника нет на месте.

В первом режиме должна отображаться вся информация о кабинете и сотруднике, который находится на своем рабочем месте: номер аудитории, к какой отделу/кафедре относится аудитория, ФИО должностного лица, его статус в настоящее время и информация о должностном лице.

В блок, отвечающий за вывод статуса, передаются два значения, первое это состояние статуса, например, «занят», «на месте» и т.д., второе значение отвечает за дополнительную информацию. Она позволяет отобразить дополнительную информацию к некоторому состоянию, например, к состоянию «нет на месте» можно добавить информацию, что пользователь находится в другом месте и придет обратно через некоторое время.

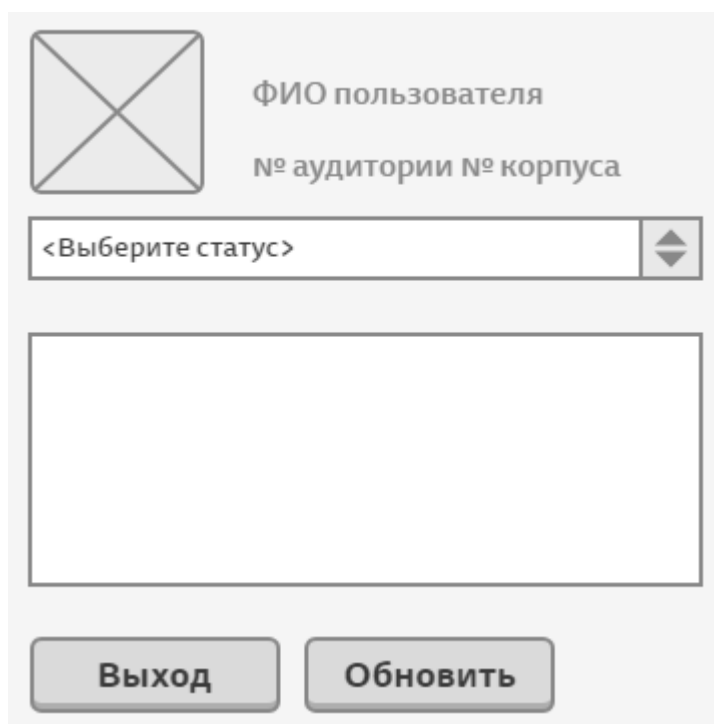
Во втором режиме работы дисплея, должно отображаться только информация о кабинете, номер кабинета и к какому отделу/кафедре относится.

## 2.5. Проектирование клиентской части

Клиентская часть была разделена на 2 части: личный кабинет сотрудника и страница администратора.

Вход на эти страницы должен осуществляться через страницу авторизации. Для входа в личный кабинет пользователю достаточно ввести свой логин-пароль. Чтобы осуществить вход от имени администратора, требуется ввод специального логин/пароля. Данный метод был выбран потому, что избавляет от создания еще лишней страницы авторизации для администратора.

На рисунке 14 отображен разработанный макет личного кабинета пользователя.



Макет личного кабинета пользователя. В верхней части находится квадратное поле с диагональными линиями (X) и текст "ФИО пользователя". Ниже него — текстовое поле "№ аудитории № корпуса". Под этими полями — выпадающий список с надписью "<Выберите статус>" и стрелками вверх/вниз. В центре — большое пустое текстовое поле. В нижней части — две кнопки: "Выход" и "Обновить".

Рисунок 14. Макет личного кабинет пользователя

Когда на сайте авторизован пользователь, в веб-браузере будут отображены следующие элементы: ФИО, номер аудитории, выпадающий список с состояниями статуса, поле для ввода дополнительной информации.

Соответственно, режим работы дисплея для преподавательских аудиторий переходит из «Пользователя нет на месте» в «Пользователь на месте» и вся информация выводится на дисплей. Поле статуса после авторизации находится в состоянии «На месте»

По нажатию кнопки «Обновить», выбранный статус и введенная дополнительная информация будут обновлены на дисплеи. По завершению рабочего дня, пользователь выходит из системы, и дисплей снова переходит в режим «Пользователя нет на месте».

При входе в систему под именем администратора, происходит перенаправление на главную страницу панели управления.

В ней была реализована система администрирования, в которой доступен ряд функций:

- просмотр персональных данных пользователей;
- редактирование;
- удаление пользователя из системы;
- возможность добавления пользователя;
- поиск по сотруднику с возможностью перехода на его профиль, либо удаление его удаление;
- просмотр любого дисплея.

На странице просмотра персональных данных, администратор сможет просмотреть всю информацию, касающуюся конкретного пользователя. Доступны операции «Редактировать» и «Удалить пользователя».

В режиме редактирования данных была разработана форма, в которой подготовлены следующие поля для заполнения: логин, пароль, имя, фамилия, отчество, должность и аудитория. При нажатии «Обновить», клиент-серверное приложение отправляет запрос на сервер на обновление данных пользователя в таблиц users с его id.

При выборе «Удалить пользователя», вся информация о пользователе удаляется из БД и происходит перенаправление на главную страницу.

При переходе на страницу «Добавить», отображается форма добавления нового пользователя. Как и в форме редактирования данных, доступны те же самые поля для заполнения. После нажатия «Добавить», приложения формирует запрос на сервер на добавление новой записи в таблицу users.

На странице просмотра дисплея было доступно 2 поля для заполнения

### 3. РЕАЛИЗАЦИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ

Перед началом реализации клиент-серверного приложения необходимо было определить этапы реализации. Так как информационная система состоит из двух частей, надо было разделить разработку приложения на 2 части.

Для каждой части проекта были созданы папки, где хранились файлы-исходники проекта.

Имя файла	Размер	Тип файла
..		
admin		Папка с файлами
config_files		Папка с файлами
css		Папка с файлами
fonts		Папка с файлами
images		Папка с файлами
jquery		Папка с файлами
js		Папка с файлами
pages		Папка с файлами
rasp		Папка с файлами
scripts		Папка с файлами
test		Папка с файлами
admin.php	3 351	Файл "PHP"
index.php	2 194	Файл "PHP"

Рисунок 15. Файловая структура информационной системы

#### 3.1. Разработка системы для статуса кабинета/аудитории

Первым было решено реализовать алгоритм работы для отображения статуса аудитории/кабинета.

Реализация данной системы была разделена на следующие этапы:

- Верстка страницы, которая отображается на дисплее;
- Написание обработчика запроса.

Для верстки страниц использовались интернет-технологии HTML и CSS с использованием Фреймворка bootstrap, что позволило лучше оформить данную страницу. Работа данного приложения со стороны сервера была написана на серверном языке программирования PHP.

На этапе проектирования было указано, что дисплеи данной системы имеют 3 состояния, соответственно, были написаны 3 веб-страницы, которые соответствовали данным состояниям.

Результаты верстки предоставлены на рисунке 16, 17, 18.

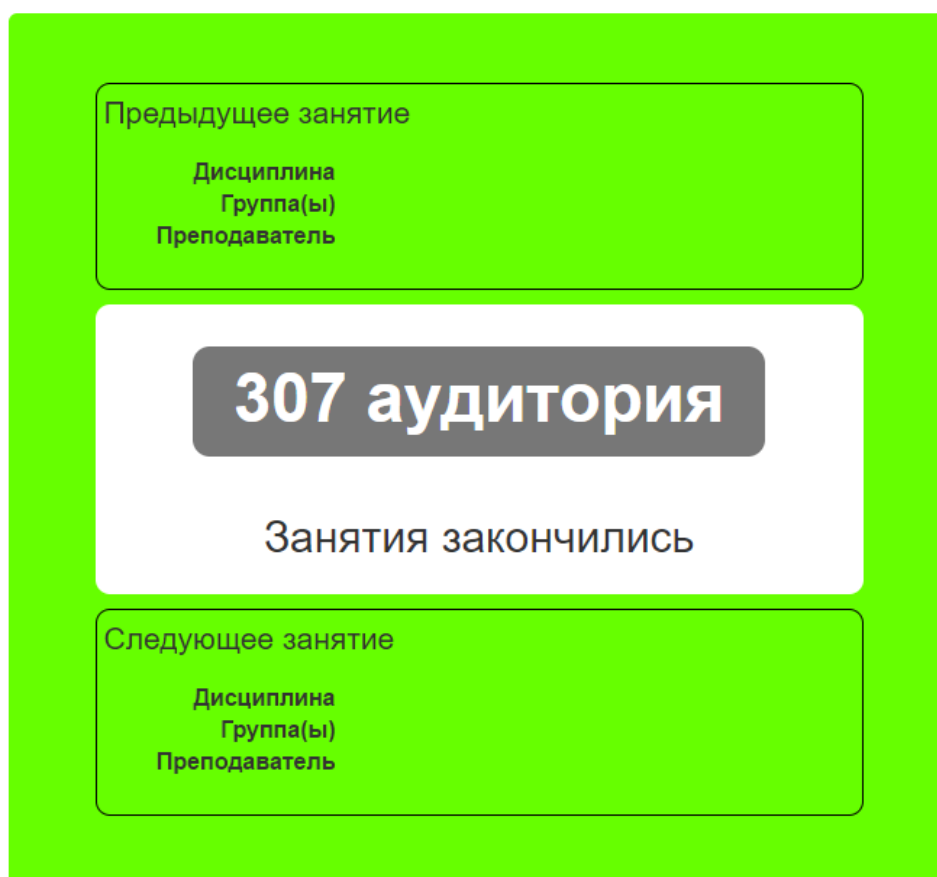


Рисунок 16. Дисплей для аудиторий/кабинетов, состояние «занятия закончились»

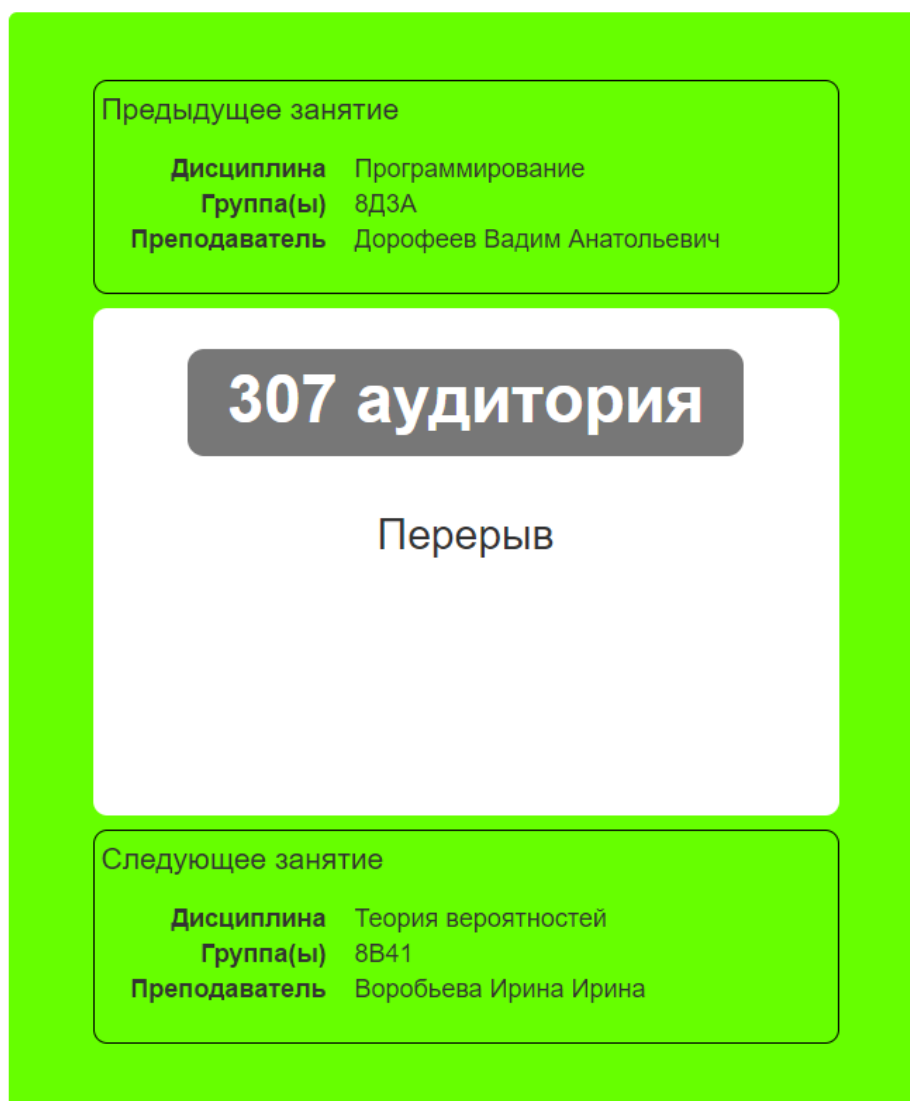


Рисунок 17. Дисплей для аудиторий/кабинетов, состояние «перерыв»

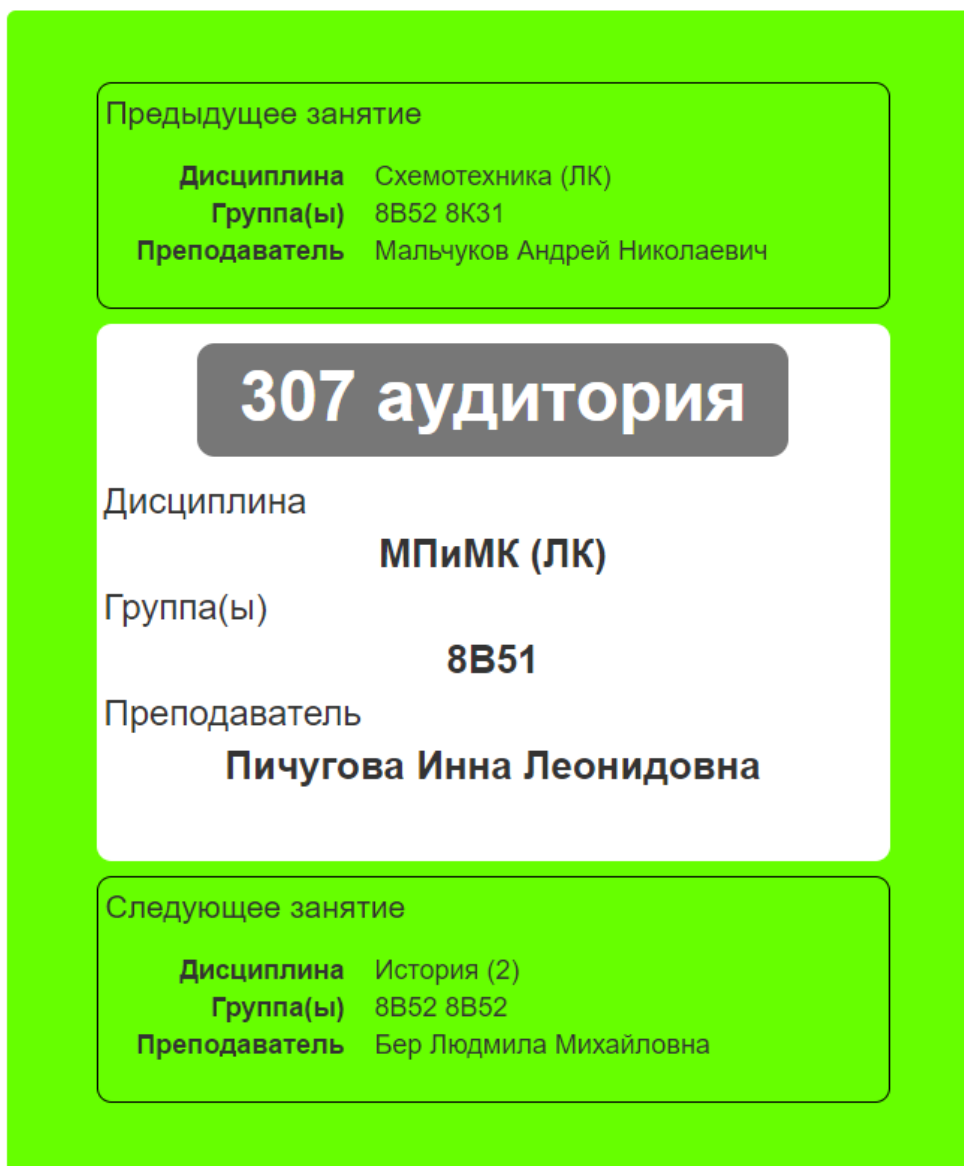


Рисунок 18. Дисплей для аудиторий/кабинетов

Запрос вывода расписания на тонком клиенте осуществляется через URL запрос. Запрос выполняется после ввода основного адреса [art.fas.aics.ru](http://art.fas.aics.ru). В конце адрес URL ставится знак «?», после него вводится запрос на требуемую аудиторию: `?id_class=307&id_build=10`. Данный запрос отправит на веб-сервер типа: «вывести информацию об аудитории с номером 307 в 10 корпусе».

Клиент-серверное приложение, получив запрос, формирует SQL-запрос на сервер к базе данных на поиск `id` данной аудитории/кабинета, по которому оно будет обращаться к таблице `gasr` для получения расписания для данного `id`.

Если есть совпадения, то приложение вернет массивы данных, в котором будет содержаться необходимая информация для данной аудитории/кабинета.

Логика работы данной системы показана на рисунке 19.

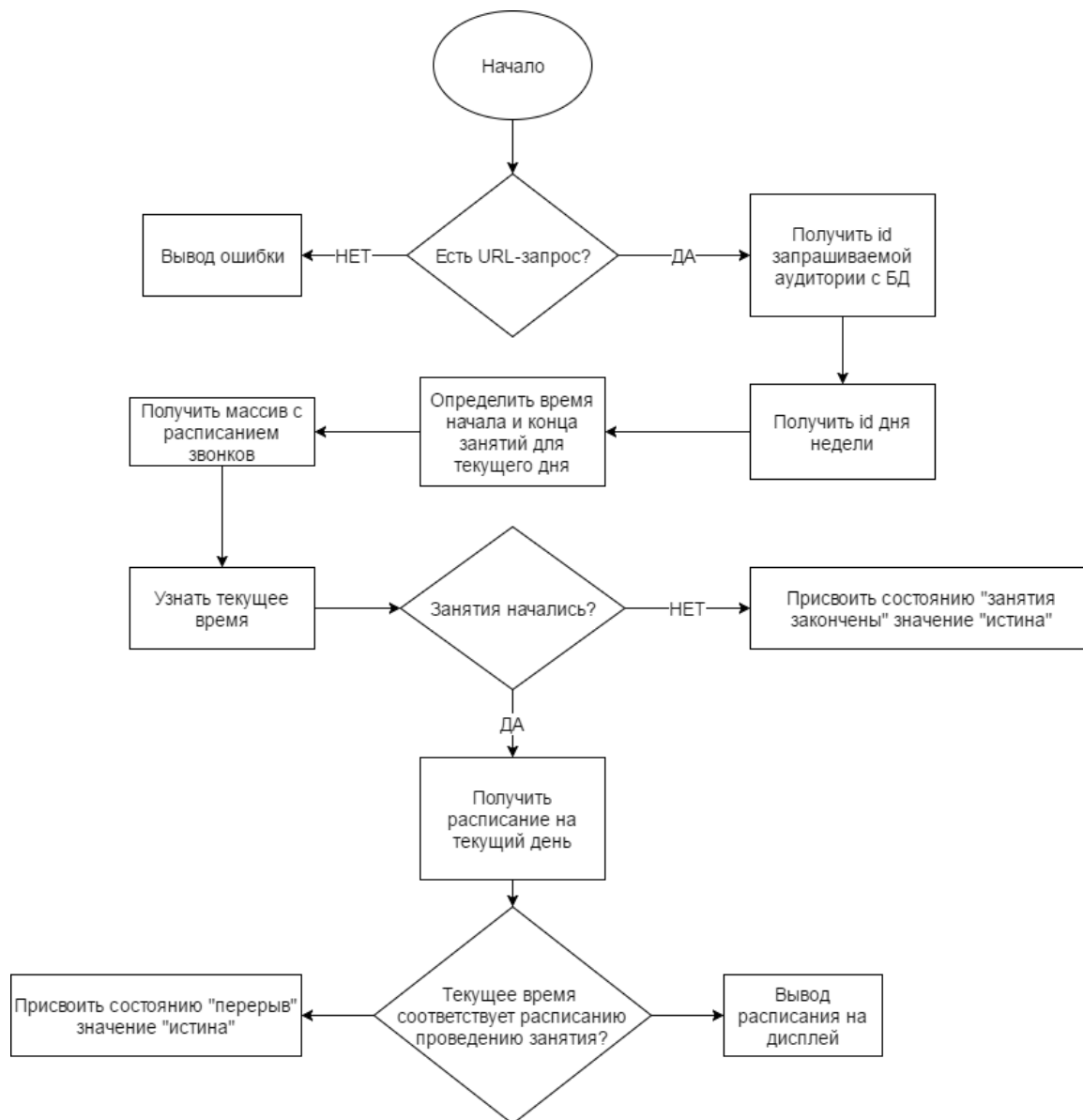


Рисунок 19. Блок-схема работы приложения

Листинг скрипта, отвечающего за выборку данных для расписания, предоставлен в приложении А.

### 3.2. Разработка системы для статуса должностного лица

Следующая система, которая была реализована – это для просмотра статуса должностного лица.

Реализация данной части приложения было также разделено на этапы:

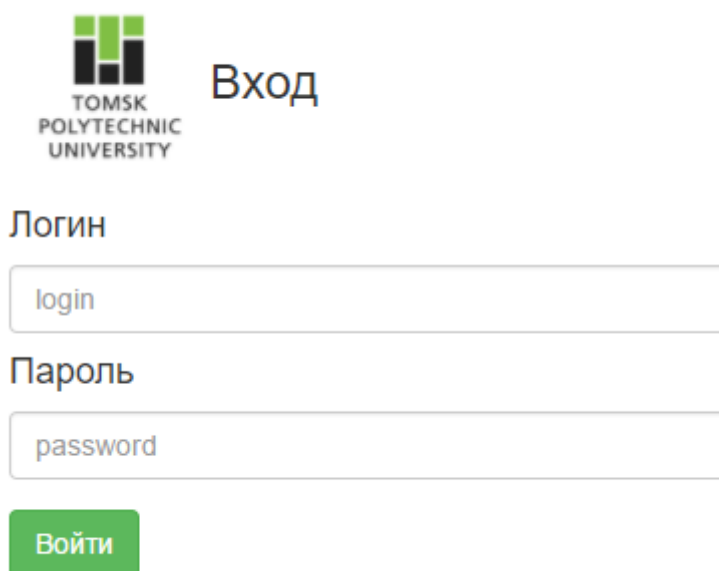


- Написания страницы авторизации;
- Написание личного кабинета;
- Написание страницы, которая выводится на дисплей;
- Написание функционала приложения;
- Написание панели управления администратора.

Как и при верстке страниц для статуса аудитории/кабинета, для страниц авторизации, личного кабинета и страницы для вывода на дисплей использовались интернет-технологии HTML и CSS с использованием Фреймворка bootstrap. Данные технологии позволили красочно оформить созданные страницы и

Функционал работы приложения со стороны сервера для данной системы был написан на серверном языке программирования PHP.

Страница авторизации – форма с полями ввода для логина и пароля должностного лица.



The image shows a login interface for Tomsk Polytechnic University. It includes the university's logo, the word 'Вход' (Login), and two input fields for 'Логин' (Login) and 'Пароль' (Password). A green 'Войти' (Login) button is positioned below the password field.

Рисунок 20. Страница авторизации

Соответственно, когда сотрудник не авторизован, дисплей отображает только номер аудитории и сообщение, что пользователь не авторизован и имеет следующий вид, как показано на рисунке 21.

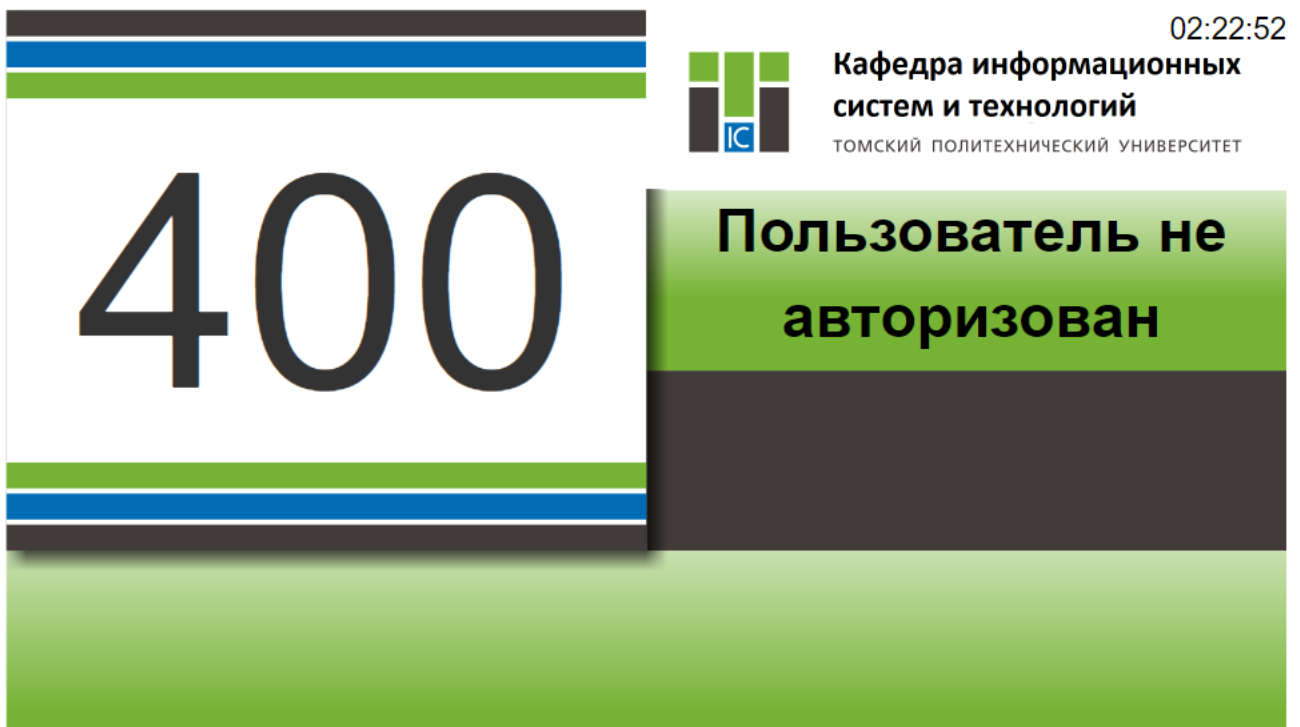
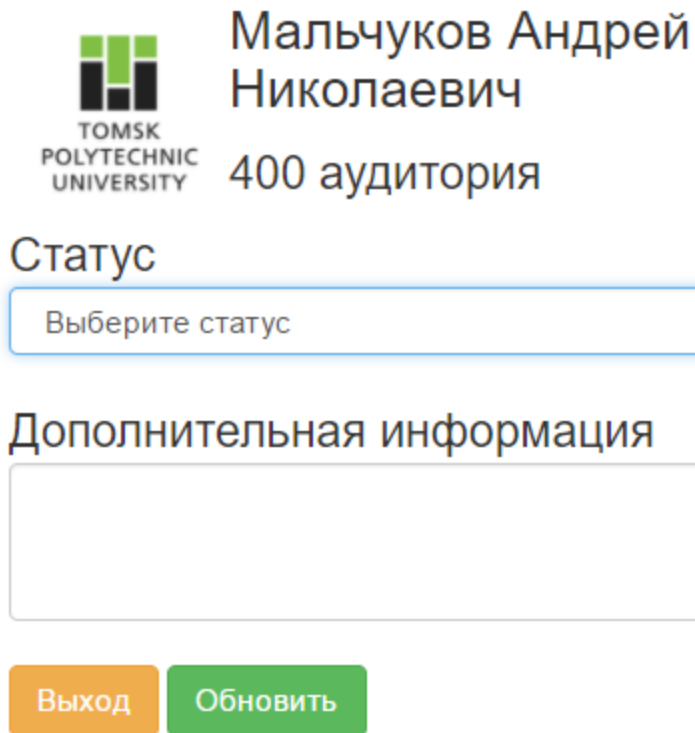



Рисунок 21. Дисплей таблички до авторизации

После авторизации пользователя, система направляет его в личный кабинет, где осуществляется управление вывода статуса на дисплей. Табличка так же обновляется в соответствии с именем сотрудника, за кем закреплена аудитория.

Сотруднику в личном кабинете показано его ФИО, и какой кабинет закреплён за ним. Пример отображаемой информации личного кабинета показан на рисунке 22.



 Мальчуков Андрей  
Николаевич  
400 аудитория

Статус  
Выберите статус ▾

Дополнительная информация

Выход Обновить

Рисунок 22. Личный кабинет пользователя

Пример отображаемой информации на табличке после авторизации показан на рисунке 23.



Рисунок 23. Дисплей таблички после авторизации

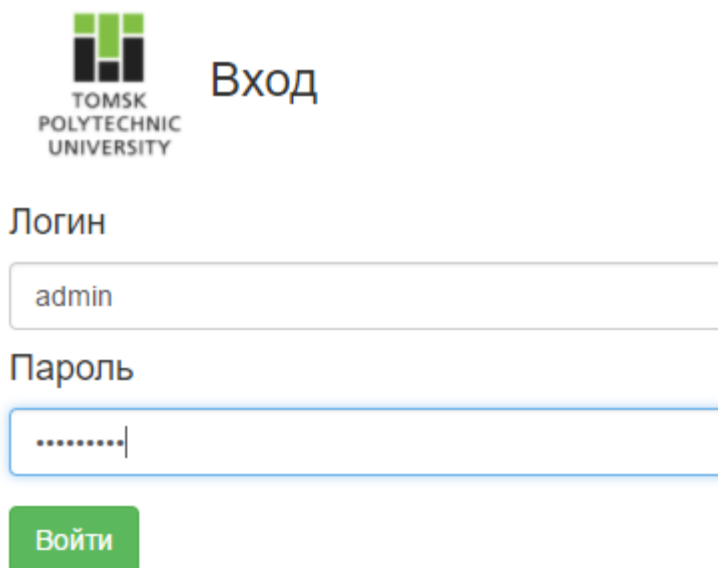
В ходе работы возник вопрос, как отслеживать сессию сотрудника на разных браузерных приложениях. Для решения данного вопроса было принято решение создать для каждого пользователя атрибут «active», который позволит отслеживать состояние пользователя.

Имеется два состояния: 1 – пользователь авторизован, 0 – пользователь не авторизован. Изначально, все пользователи находятся в состоянии «0», после авторизации, приложение формирует SQL-запрос на обновление атрибута «active» в состояние «1». Страница дисплея, перед загрузкой на табличке, проверяет данный атрибут и в зависимости от актуального состояния выводит требуемую страницу.

Листинг работы данной системы предоставлен в приложении Б.

При определенном наборе значений логин/пароль, система определит авторизующего, как администратора и перенаправит на соответствующую страницу.

Подключенный файл, отвечающий за авторизацию пользователей, проверяет комбинацию логина и пароля для входа в панель управления.



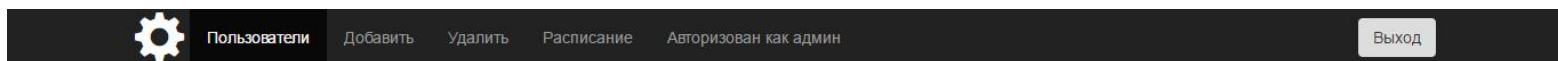
The image shows a login interface for Tomsk Polytechnic University. At the top left is the university's logo, which consists of three vertical bars of varying heights (green, black, green) above the text 'TOMSK POLYTECHNIC UNIVERSITY'. To the right of the logo is the word 'Вход' (Login) in a large, bold, black font. Below this, there are two input fields. The first is labeled 'Логин' (Login) and contains the text 'admin'. The second is labeled 'Пароль' (Password) and contains a series of dots, indicating a masked password. Below the password field is a green button with the white text 'Войти' (Login).

Рисунок 24. Вход в систему управления

### 3.3. Реализация панели управления администратора

Как было выше описано, разработанная панель управления позволяет осуществлять контроль пользователей, просмотр дисплеев каждой аудитории

а так же поиск пользователей. Далее подробно описан каждый функционал панели управления.



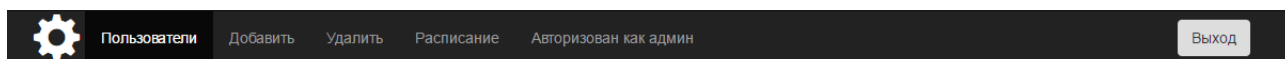
## Пользователи

-  [Фадеев Александр Сергеевич](#)
-  [Бер Людмила Михайловна](#)
-  [Коротченко Константин Борисович](#)
-  [Хабас Вера Петровна](#)
-  [Мальчуков Андрей Николаевич](#)
-  [Дорофеев Вадим Анатольевич](#)
-  [Воробьева Ирина Павловна](#)
-  [Пичугова Инна Леонидовна](#)
-  [Куликова Ольга Александровна](#)
-  [Осокин Александр Николаевич](#)

Рисунок 25. Страница управления администратора

### 3.3.1. Профиль пользователя: редактирование и удаление

Данная система предоставляет доступ к просмотру профиля пользователя системы, с возможностью редактирования его персональных данных, удаления пользователя, добавления нового пользователя в систему и также просмотр дисплеев аудиторий/кабинетов.



## Коротченко Константин Борисович

### Информация пользователя

**Логин** kkb  
**Рабочее место** 03 корпус, 206 аудитория  
**Должность** Доцент кафедры экспериментальной физики

Удалить пользователя

Редактировать

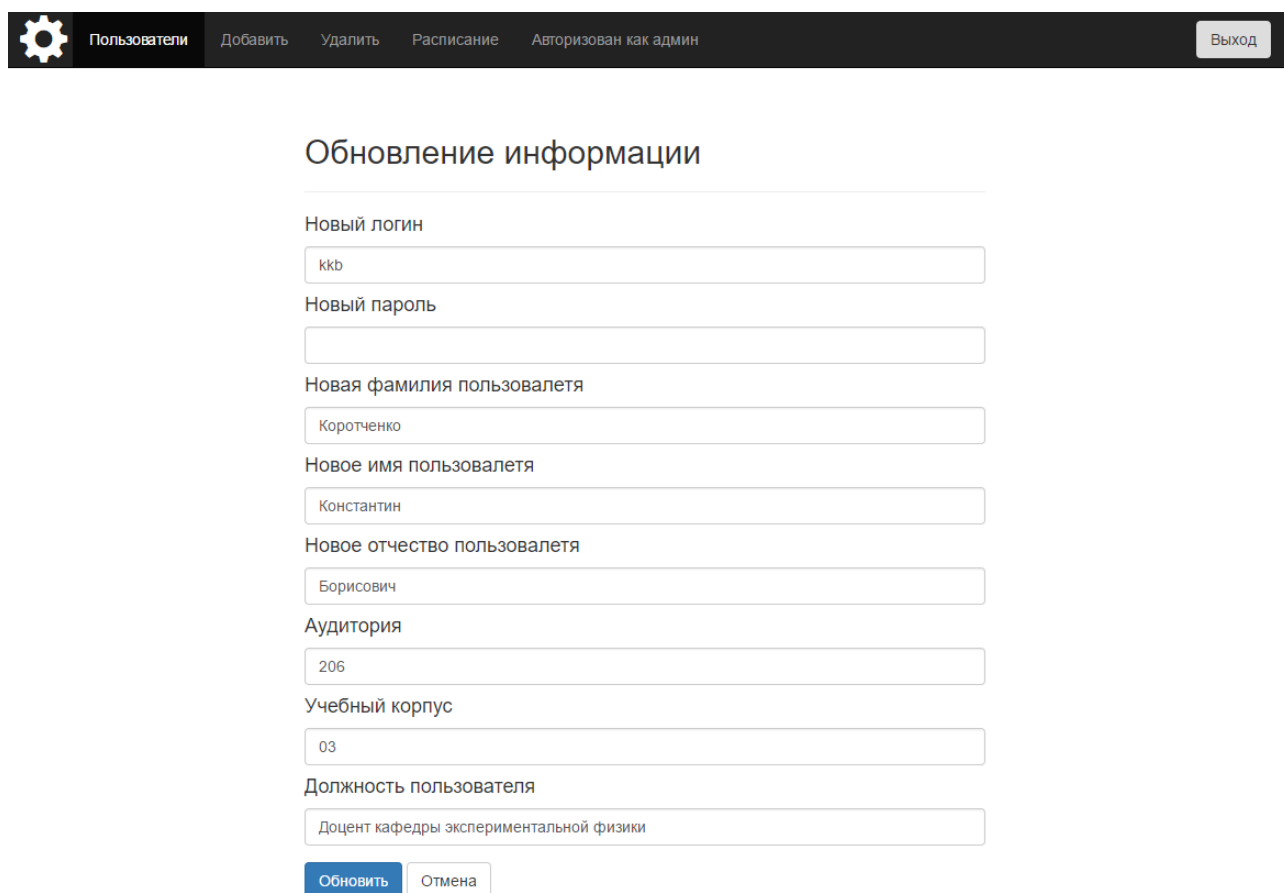
Посмотреть дисплей

Рисунок 26. Просмотр профиля пользователя

При переходе по ссылке «Редактировать» администратору открывается форма обновления информации. Форма содержит поля следующие поля для ввода:

- Новый логин;
- Новый пароль;
- Новая фамилия пользователя;
- Новое имя пользователя;
- Новое отчество пользователя;
- Аудитория;
- Учебный корпус;
- Должность пользователя.

Данная форма показана на рисунке 27.



The screenshot shows a web application interface with a dark header bar. On the left, there is a gear icon and the text 'Пользователи'. In the center, there are navigation links: 'Добавить', 'Удалить', 'Расписание', and 'Авторизован как админ'. On the right, there is a 'Выход' button. The main content area is titled 'Обновление информации' and contains several input fields with labels and pre-filled values: 'Новый логин' (kkb), 'Новый пароль' (empty), 'Новая фамилия пользователя' (Коротченко), 'Новое имя пользователя' (Константин), 'Новое отчество пользователя' (Борисович), 'Аудитория' (206), 'Учебный корпус' (03), and 'Должность пользователя' (Доцент кафедры экспериментальной физики). At the bottom of the form are two buttons: 'Обновить' and 'Отмена'.

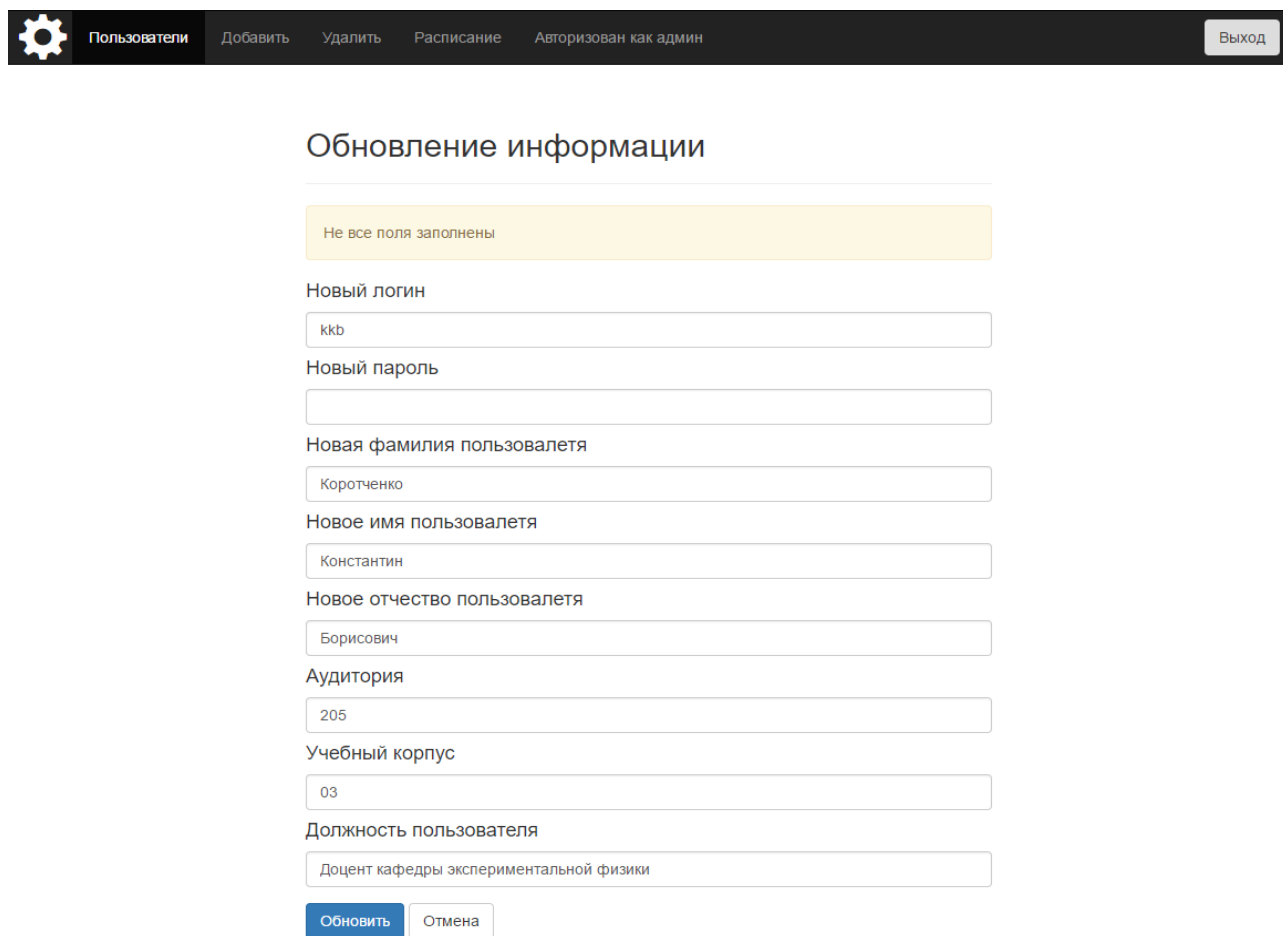
Рисунок 27. Форма редактирования информации

Изначально все поля, кроме поля «Новый пароль» уже заполнены теми значениями из таблицы users, которые имеются в БД. Это сделано в случае, если некоторые данные надо оставить без изменения.

Если все поля изменены, то после нажатия на кнопку «Обновить», приложение формирует SQL-запрос типа «UPDATE SET», который обновляет запись пользователя в таблице users с заданным id.

Что касается поля «Новый пароль», то оно остается пустым, так как в БД оно хранится в зашифрованном виде и выводить его нет смысла. Если нет надобности в обновлении пароля, то данное поле можно оставить пустым. Тогда приложение сформирует SQL-запрос, не включая в него атрибут «pass» и значение атрибута в БД останется тем же.

Если SQL-запрос не прошел успешно, то выведется сообщение «Не все поля заполнены».



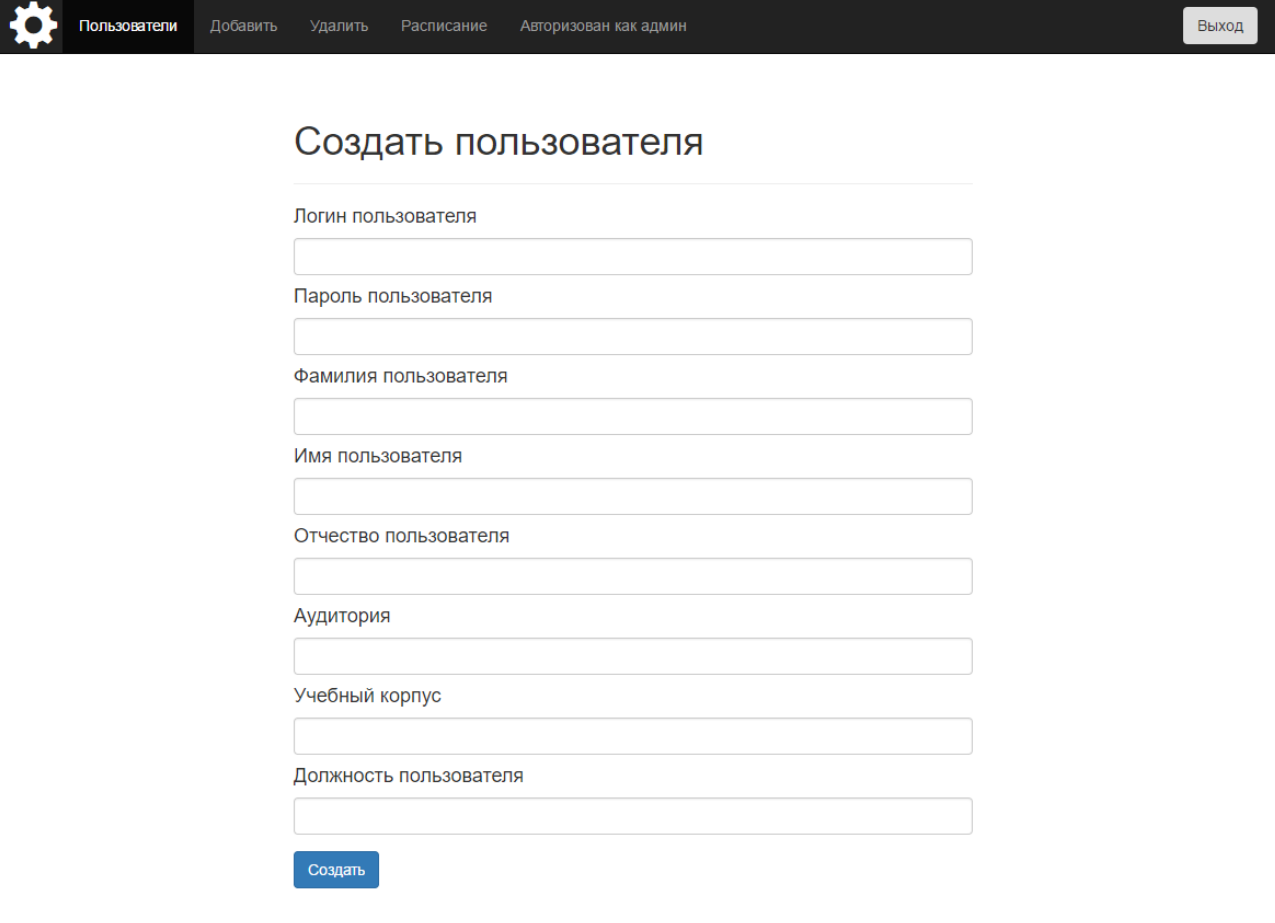
The screenshot shows a web interface for updating user information. At the top, there is a navigation bar with a gear icon, the text 'Пользователи', and several action buttons: 'Добавить', 'Удалить', 'Расписание', 'Авторизован как админ', and a 'Выход' button. The main content area is titled 'Обновление информации'. A yellow error message box at the top of the form reads 'Не все поля заполнены'. Below this, the form has several input fields: 'Новый логин' (k kb), 'Новый пароль' (empty), 'Новая фамилия пользователя' (Коротченко), 'Новое имя пользователя' (Константин), 'Новое отчество пользователя' (Борисович), 'Аудитория' (205), 'Учебный корпус' (03), and 'Должность пользователя' (Доцент кафедры экспериментальной физики). At the bottom of the form are two buttons: 'Обновить' and 'Отмена'.

Рисунок 28. Ошибка обновления информации

Листинг приложения, реализующий функционал данной части данного пункта, предоставлен в приложении В

### 3.3.2. Добавить пользователя

При переходе на страницу «Добавить» отображается форма аналогичная форме редактирования персональной информации пользователя.



The screenshot shows a web application interface with a dark navigation bar at the top. The bar contains a gear icon, the text 'Пользователи', and several menu items: 'Добавить', 'Удалить', 'Расписание', and 'Авторизован как админ'. A 'Выход' button is located on the right side of the bar. Below the navigation bar, the main content area is titled 'Создать пользователя'. It features a form with the following fields: 'Логин пользователя', 'Пароль пользователя', 'Фамилия пользователя', 'Имя пользователя', 'Отчество пользователя', 'Аудитория', 'Учебный корпус', and 'Должность пользователя'. Each field is represented by a white input box with a light gray border. At the bottom of the form is a blue button labeled 'Создать'.

Рисунок 29. Страница «Добавить»

Заполнив все поля и нажав кнопку «Создать», приложение формирует SQL-запрос на создание новой записи в таблице users. При успешном запросе, будет выведено сообщение «Пользователь добавлен».



## Создать пользователя

Пользователь добавлен

Логин пользователя

Пароль пользователя

Фамилия пользователя

Имя пользователя

Отчество пользователя

Аудитория

Учебный корпус

Должность пользователя

Создать

Рисунок 30. Удачное добавление пользователя

Если же какое-то поле не заполнено, то будет выведено сообщение «Ошибка добавления пользователя».



## Создать пользователя

Ошибка добавления пользователя

Логин пользователя

Пароль пользователя

Фамилия пользователя

Имя пользователя

Отчество пользователя

Аудитория

Учебный корпус

Должность пользователя

Создать

Рисунок 31. Ошибка добавления нового пользователя

Листинг приложения, реализующий функционал данного пункта, предоставлен в приложении Г.

### 3.3.3. Поиск пользователей в системе

В основе работы данного поиска использовался технология live search, который используется поисковым сервисом Google, то есть результаты запроса выводятся без обновления страницы. Возможность такого вывода информации дает использование JavaScript и AJAX. Сам запрос на получения данных пользователя осуществляется скриптом, написанный на PHP. Приложение также формирует SQL-запрос «SELECT FROM». Результат запроса будет выводить все совпадения, не зависимо от регистра или языка ввода и при вводе одного символа в строку поиска. Такой поиск осуществляется за счет использования регулярных выражений. Регулярные

выражения являются своеобразным языком работы с текстом и для осуществления манипуляций с подстроками в тексте.

Результатом поиска является таблица со списком записей, которые удовлетворяют запросу в поисковой строке. Таблица состоит из четырех атрибутов, Фамилия, Имя, Отчество, Профиль.

В атрибуте «Профиль» расположена кнопка-ссылка на профиль пользователя. Такая система облегчит поиск пользователя и обеспечивает динамическое отображение результатов.

Листинг работы данного поиска предоставлен в приложении Д.

Пользователи    Добавить    Поиск    Расписание    Авторизован как админ    Выход

### Поиск по пользователю

Поиск а

Фамилия	Имя	Отчество	Профиль
Фадеев	Александр	Сергеевич	<a href="#">Профиль</a>
Бер	Людмила	Михайловна	<a href="#">Профиль</a>
Коротченко	Константин	Борисович	<a href="#">Профиль</a>
Хабас	Вера	Петровна	<a href="#">Профиль</a>
Мальчуков	Андрей	Николаевич	<a href="#">Профиль</a>
Дорофеев	Вадим	Анатолевич	<a href="#">Профиль</a>
Воробьева	Ирина	Павловна	<a href="#">Профиль</a>
Пичугова	Инна	Леонидовна	<a href="#">Профиль</a>
Куликова	Ольга	Александровна	<a href="#">Профиль</a>
Осокин	Александр	Николаевич	<a href="#">Профиль</a>
Сидоров	Владислав	Викторович	<a href="#">Профиль</a>

Рисунок 32. Поиск пользователя

### 3.4. Настройка и работа системы на платформе Raspberry PI

После реализации функционала клиент-серверного приложения, данная система была запущена на Raspberry PI.

На данной платформе был запущен браузер, в адресную строку был записан URL запрос, описанный ранее.

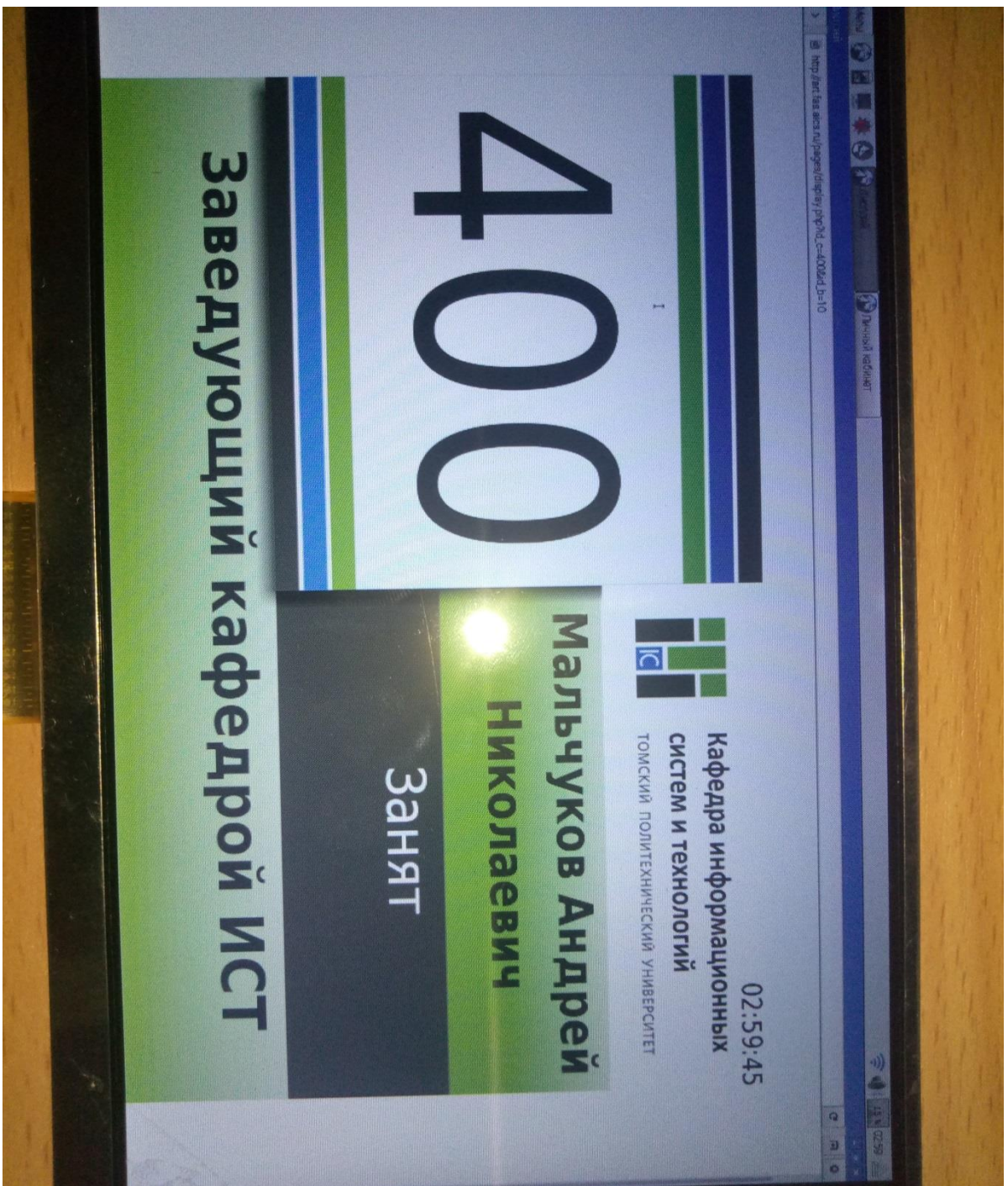


Рисунок 33. Система статуса должностного лица

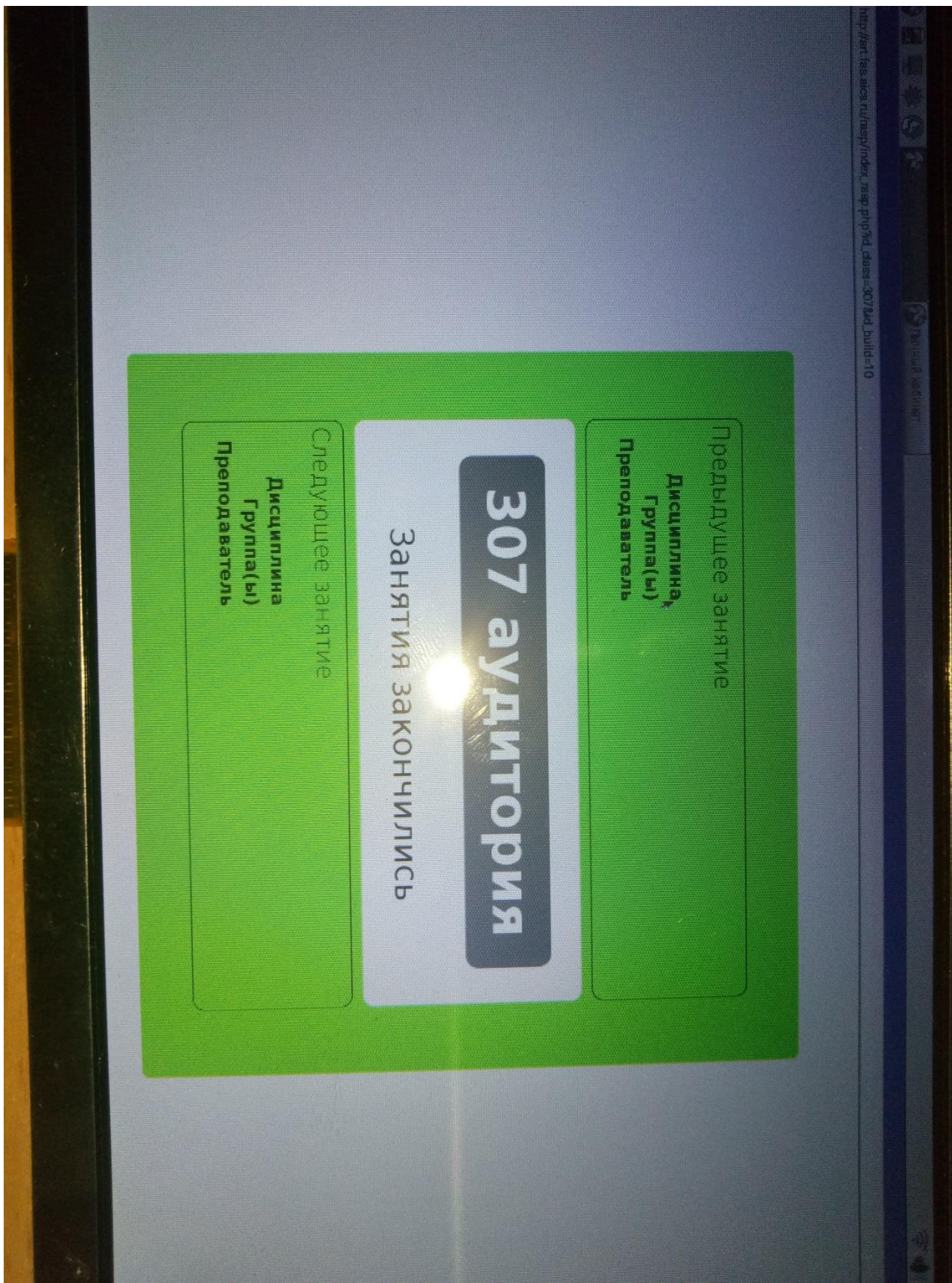


Рисунок 34. Система статуса кабинета/аудитории

## ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа	ФИО
8В3А	Чайбар Артышу Алимовичу

Институт	ИК	Кафедра	ИСТ
Уровень образования	бакалавриат	Направление/специальность	09.03.01 «Информатика и вычислительная техника»

### Исходные данные к разделу «Социальная ответственность»:

*Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения*

*Программное обеспечение*  
*Разработанное устройство может использоваться в следующих структурах:*

- *в офисных помещениях,*
- *для оперативного информирования о статусе должностных лиц в кабинетах,*
- *в учебных заведениях для информации и занятии, проводимом в аудитории,*
- *в медицинских учреждениях, которые предъявляют повышенные требования в оперативной информации о том или ином кабинете для посетителей,*
- *в социальных и сервисных службах, где требуется упорядочивание посещения кабинетов посетителями и др.*

### Перечень вопросов, подлежащих исследованию, проектированию и разработке:

#### **1. Производственная безопасность**

*1.1. Анализ выявленных вредных факторов при разработке и эксплуатации проектируемого решения в следующей последовательности:*

- *физико-химическая природа вредности, её связь с разрабатываемой темой;*
- *действие фактора на организм человека;*
- *приведение допустимых норм с необходимой размерностью (со ссылкой на соответствующий нормативно-технический документ);*
- *предлагаемые средства защиты;*
- *(сначала коллективной защиты, затем – индивидуальные защитные средства).*

*1.2. Анализ выявленных опасных факторов при разработке и эксплуатации проектируемого решения в следующей последовательности:*

- *механические опасности (источники, средства защиты);*
- *термические опасности (источники, средства защиты);*
- *электробезопасность (в т.ч. статическое электричество, молниезащита – источники, средства защиты)*

*Вредные:*

8. *повышение уровней шума и вибраций;*
  9. *недостаточная освещенность рабочей зоны;*
  10. *отклонение параметров микроклимата в помещении;*
  11. *уровень шума на рабочем месте; излучение электромагнитных полей*
- Опасные:*

12. *электрический ток;*
13. *пожарная безопасность;*
14. *повышенная запыленность и загазованность воздуха рабочей зоны.*

<p><b>2. Экологическая безопасность:</b></p> <ul style="list-style-type: none"> <li>• защита селитебной зоны анализ воздействия объекта на атмосферу (выбросы);</li> <li>• анализ воздействия объекта на гидросферу (сбросы);</li> <li>• анализ воздействия объекта на литосферу (отходы);</li> <li>• разработать решения по обеспечению экологической безопасности со ссылками на НТД по охране окружающей среды.</li> </ul>	<p>Утилизация используемой орг.техники и люминесцентных ламп.</p>
<p><b>3. Безопасность в чрезвычайных ситуациях:</b></p> <ol style="list-style-type: none"> <li>1. перечень возможных ЧС при разработке и эксплуатации проектируемого решения;</li> <li>2. выбор наиболее типичной ЧС;</li> <li>3. разработка превентивных мер по предупреждению ЧС;</li> <li>4. разработка действий в результате возникшей ЧС и мер по ликвидации её последствий.</li> </ol>	<p>Чрезвычайная ситуация техногенного характера для данного помещения – пожар. Установа общих правил поведения и рекомендаций во время пожара, план эвакуации.</p>
<p><b>4. Правовые и организационные вопросы обеспечения безопасности:</b></p> <ol style="list-style-type: none"> <li>5. специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства;</li> <li>6. организационные мероприятия при компоновке рабочей зоны.</li> </ol>	<p>Основные проводимые правовые и организационные мероприятия по обеспечению безопасности трудящихся в учебных аудиториях.</p>
<p><b>Перечень графического материала:</b></p>	
<p>При необходимости представить эскизные графические материалы к расчётному заданию (обязательно для специалистов и магистров)</p>	

<p><b>Дата выдачи задания для раздела по линейному графику</b></p>	
--	--

**Задание выдал консультант:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Инженер	Малана Наталья Викторовна	К.Т.Н.		

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8В3А	Чайбар Артыш Алимович		

## 4. СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ

В любой научно-исследовательской и проектной деятельности немаловажную роль занимает такая область как безопасность труда и окружающей среды.

В понятие «социальная ответственность» входит следующее: состояние рабочего места, помещения, режим трудовой деятельности и обеспечение мероприятий по защите трудящихся в моменты чрезвычайных ситуаций регламентируются в соответствии с международным стандартом ICCSR26000:2011 «Социальная ответственность организации» [13]. Целью данного стандарта является принятие проектных решений, исключающих несчастные случаи на производстве и снижение негативных воздействий на окружающую среду.

Согласно данному стандарту такое понятие, как «социальная ответственность», означает ответственность организации за воздействие решений, которые были ею предложены, на общество и окружающую среду.



#### **4.1. Введение**

В данной выпускной квалификационной работе создавались информационные интерактивные таблички аудиторий, которые отображают статус учебной аудитории/кабинета или статус должностного лица.

Цель работы – создание распределенной информационной системы, отображающей на дверных табличках-дисплеях информацию о текущем статусе должностных лиц в кабинете, мероприятии, проходящем в аудитории, текущем занятии в учебных аудиториях с уточнением наименования учебной дисциплины, имени преподавателя и номеров учебных групп.

Разработанное устройство может использоваться в следующих структурах:

- в офисных помещениях,
- для оперативного информирования о статусе должностных лиц в кабинетах,
- в учебных заведениях для информации и занятия, проводимом в аудитории,
- в медицинских учреждениях, которые предъявляют повышенные требования в оперативной информации о том или ином кабинете для посетителей,
- в социальных и сервисных службах, где требуется упорядочивание посещения кабинетов посетителями и др.

Исходя из выше перечисленного списка использования данного устройства, пользователями могут быть: работники, сотрудники, учащиеся учебных заведений, посетители пациентов и т.д.

## 4.2. производственная безопасность

Отрицательные факторы предполагают большой объем работы с ПК, поэтому важным критерием безопасности является организация рабочего места и режима трудовой деятельности.

Таблица 17. Опасные и вредные факторы при написании информационной системы [14]

Источник фактора, наименование видов работ	Факторы (по ГОСТ 12.0.003-74)		Нормативные документы
	Вредные	Опасные	
Работы в помещении: Создание информационной системы	15. недостаточная освещенность рабочей зоны; 16. отклонение параметров микроклимата в помещении; 17. уровень шума на рабочем месте; 18. излучение электромагнитных полей	19. электрический ток; 20. пожарная безопасность; 21. повышенная запыленность и загазованность воздуха рабочей зоны.	

Выпускная квалификационная работа студента выполнялась в десятом корпусе ТПУ на кафедре информационных систем и технологий. Рабочее место находится на четвертом этаже здания и представляет собой комнату длиной – 5 м., шириной – 4 м. и высотой – 3 м. Естественное освещение кабинета осуществляется посредством одного окна размерами 2,2 м. х 1,5 м. Дверь – металлическая, одностворчатая, черного цвета. Высота двери – 2 м., ширина - 1 м. Стены комнаты окрашены водоэмульсионной краской бежевого цвета. Потолок подвесной, плиточный. Пол покрыт линолеумом. Площадь кабинета составляет 20 м<sup>2</sup>, объем – 60 м<sup>3</sup>.

Согласно СанПиН 2.2.2/2.4.1340-03 [15], норма площади рабочего места с персональным компьютером составляет 4,5 м<sup>2</sup>. В рассматриваемой аудитории установлено 4 рабочих места с персональными компьютерами и жидкокристаллическими экранами. Соответственно, на одного человека приходится 5 м<sup>2</sup>, что соответствует вышеуказанным требованиям.

### 4.2.1. Освещенность рабочей зоны

Рабочее (общее) освещение – это основное освещение, обеспечивающее нормальные условия для нахождения человека в помещении. Под нормальными понимаются условия жизнедеятельности человека, при которых он не напрягает зрение, чтобы выполнить любое действие, для которого данное помещение предназначено. [16].

Освещение в недостаточной степени может привести к напряжению зрения, ослаблению внимания и наступлению преждевременной утомленности. Слепление, резь в глазах и раздражение могут быть вызваны чрезмерно ярким освещением. Свет на месте труда может создать сильные тени или отблески, а также дезориентировать работающего. Основным документом, регламентирующим нормы освещенности, является СНиП 23-05-95 [16].

Основным показателем качества освещения является освещенность  $E$  - поверхностная плотность светового потока. По характеристике зрительной работы труд программиста относится к разряду III подразряду Г (высокой точности), т.е. наименьший размер объекта различения от 0,3 до 0,5 мм (точка) [3]. Это значит, что нормативное значение освещенности рабочего места должно быть 200 лк (СНиП 23-05-95) [16].

Рассчитаем фактическую освещенность рассматриваемой учебной аудитории. Длина и ширина аудитории равны соответственно 5 и 4 м, высота – 3 м. Рассчитаем индекс помещения:

$$i = \frac{S}{h*(A+B)} ,$$

где  $i$  – индекс помещения;

$S$  – площадь помещения, м<sup>2</sup>;

$h$  – высота помещения, м;

$A$  – длина помещения, м;

$B$  – ширина помещения.

$$i = \frac{20}{3*(5+4)} = 0.7,$$

Исходя из значения индекса помещения, можно определить, что коэффициент использования рассматриваемого светового светильника с люминесцентными лампами равен 26% [17].

Рассчитаем освещенность по формуле, учитывая, что в аудитории 4 светильника по 4 лампы в каждом:

$$E_{\text{факт}} = \frac{N * n * \Phi_{\text{ст}} * \eta}{S * K_3 * Z} ,$$

где  $E_{\text{н}}$  – фактическая освещенность;

$N$  – число светильников в помещении;

$n$  – число ламп в светильнике;

$\Phi_{\text{ст}}$  – величина стандартного светового потока, лм;

$\eta$  – коэффициент использования светового потока;

$S$  – площадь помещения;

$K_3$  – коэффициент запаса;

$Z$  – коэффициент неравномерности освещения.

Зная, что  $\Phi_{\text{ст}} = 1450$  лм для люминесцентных ламп дневной света ЛБЦ-30 (СНиП 23-05-95),  $K_3$  для помещений с малым выделением пыли равен 1,5, а  $Z$  для люминесцентных ламп равен 1 рассчитаем значение фактической освещенности.

$$E_{\text{факт}} = \frac{4 * 4 * 1450 * 0,26}{20 * 1,5 * 1} = 201,1 \text{ лк} ,$$

Рассчитаем численную оценку разности между фактическим значением освещенности и нормативным.

$$\Delta E = \frac{(E_{\text{факт}} - E_{\text{н}})}{E_{\text{н}}} * 100\% ,$$

где  $\Delta E$  – показатель разности между фактической освещенностью и нормативной;

$E_{\text{факт}}$  – фактическое значение освещенности;

$E_{\text{н}}$  – нормативное значение освещенности.

$$\Delta E = \frac{(201,1 - 200)}{200} * 100\% = 0,5\%$$

Отсюда можно заключить, что в аудитории подходящая система освещения, так как сохраняется допустимое отклонение освещенности в 20% [16].

#### **4.2.2. Производственный шум**

Люди, которым приходится работать в условиях длительного шума, обычно имеют головные боли, раздражительность, сталкиваются со снижением памяти, повышенной утомляемостью, также у многих понижен аппетит, есть боли в ушах и т. д. Перечисленные факты снижают производительность, работоспособность человека, а также качество труда [18].

Шумовой фон помещения создают десять одновременно работающих компьютеров. Также возникает шум, исходящий от принтера или телефонных аппаратов. Также источником шума является система вентиляции или шумы, поступающие извне помещения.

Во избежание негативных последствий от производственного шума, его необходимо регулировать в соответствии с нормами, которые указаны в ГОСТ 12.1.003-83 «ССБТ. Общие требования безопасности» [19].

Допустимые уровни звука и звукового давления для рабочего места разработчика-программиста согласно вышеуказанному ГОСТу 12.1.003-83 представлены в таб. 5.1.

Таблица 18 – Предельно допустимые уровни звука (ГОСТ 12.1.003-83

[19])

Вид трудовой деятельности/ Частоты	Уровни звука и звукового давления, дБ, в октавных полосах со среднегеометрическими частотами, Гц								
	1,5	3	25	50	00	000	000	000	000
Научная деятельность, проектирование, программирование, Рабочие места проектно-конструкторских бюро, программистов вычислительных машин и т.д.	6	1	1	4	9	5	2	0	8

Допустимый уровень звукового давления колеблется от 38 дБ до 86 дБ при частоте от 8000 Гц до 31,5 Гц, соответственно.

Для уменьшения воздействий шума можно использовать следующие методы, согласно СНиП 23-03-2003 [20]:

экранирование рабочих мест, то есть установка перегородок между рабочими местами;

установка оборудования, производящего минимальный шум.

Для снижения уровня шума, производимого персональными компьютерами, рекомендуется регулярно проводить их техническое обслуживание: чистка от пыли, замена смазывающих веществ; также применяются звукопоглощающие материалы.

### 4.2.3. Микроклимат помещения

Компьютеры могут привести к увеличению температуры и снижению относительной влажности в помещении. В СанПиН 2.2.4.548 – 96 установлены величины параметров микроклимата, создающие комфортные условия [21].

Работа программиста относится к легкой категории 1Б (СанПиН 2.2.4.548 – 96) [21]. В таблицах представлены данные показатели для теплого периода года (плюс 10 °С и выше) и для холодного периода года.

Таблица 19 – Оптимальные величины показателей микроклимата (СанПиН 2.2.4.548 – 96) [21]

Период года	Температура воздуха, °С	Температура поверхностей, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с
Холодный	21-23	20-24	40-60	0,1
Теплый	22-24	21-25		0,1

Таблица 20 – Допустимые величины показателей микроклимата (СанПиН 2.2.4.548 – 96) [21]

Период года	Температура воздуха, °С	Температура поверхностей, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с
Холодный	19-24	18-25	15-75	0,1-0,2
Теплый	20-28	19-29		0,1-0,3

Если температура воздуха отличается от нормальной, то время пребывания в таком помещении должно быть ограничено в зависимости от категории тяжести работ. Температура в рассматриваемом помещении в холодное время года может опускаться до 19-21 °С, а в теплое время года

подниматься до 25-28 °С. Данные показатели соответствуют допустимым значениям температуры.

Таблица 21 – Рекомендуемое время работы при температуре воздуха ниже допустимых величин (СанПиН 2.2.4.548 – 96) [21]

Температура воздуха, °С	Время пребывания, не более, ч
17	6
18	7

Таблица 22 – Рекомендуемое время работы при температуре воздуха выше допустимых величин (СанПиН 2.2.4.548 – 96) [20]

Температура воздуха, °С	Время пребывания, не более, ч
30,0	5
29,5	5,5
29,0	6

К мероприятиям по оздоровлению воздушной среды в производственном помещении относятся правильная организация вентиляции и кондиционирования воздуха, отопление помещений. В рассматриваемой аудитории вентиляция осуществляется естественным и механическим путём. В зимнее время в помещении предусматривается система отопления. Это обеспечивает нормальное состояние здоровья работников в аудитории.

#### **4.2.4. Электромагнитное излучение**

Электромагнитное излучение - распространяющееся в пространстве возмущение электрических и магнитных полей [18]. Источниками электромагнитного излучения в данном исследовании являются мониторы и системный блок.

Оценка величины уровней ЭМП, проведенная по паспортным данным компьютера и монитора, показала их соответствие нормам ТСО–03 и СанПиН 2.2.2/2.4.1340–03[13]. В табл. 6 приведены нормы уровня ЭМП, которым соответствует техника в кабинете.



Таблица 23 – Допустимые уровни ЭМП, создаваемых ПК (СанПиН 2.2.2/2.4.1340–03 [13])

Наименование параметров		ВДУ ЭМП
Напряженность электрического поля	в диапазоне частот 5 Гц - 2 кГц	25 В/м
	в диапазоне частот 2 кГц - 400 кГц	2,5 В/м
Плотность магнитного потока	в диапазоне частот 5 Гц - 2 кГц	250 нТл
	в диапазоне частот 2 кГц - 400 кГц	25 нТл
Электростатический потенциал экрана видеомонитора		500 В

Для того, чтобы снизить воздействие таких видов излучения, рекомендуют применять такие мониторы, у которых уровень излучения понижен (MPR-II, TCO-92, TCO-99), а также установить защитные экраны и соблюдать режимы труда и отдыха.

#### 4.2.5. Поражение электрическим током

К опасным факторам относят поражение электрическим током согласно ГОСТ 12.0.003-74 [23]. Персональный компьютер питается от сети 220В переменного тока с частотой 50Гц. Помещение с ПЭВМ, где проводились описанные выше работы, относится к помещениям без повышенной опасности (ГОСТ 12.1.019 [25]).

К мероприятиям по предотвращению возможности поражения электрическим током относятся:

- при включенном сетевом напряжении работы на задней панели компьютера должны быть запрещены;
- все работы по устранению неисправностей должен производить квалифицированный персонал;
- необходимо постоянно следить за исправностью электропроводки.

### 4.3. Пожарная безопасность

Также к опасным факторам относится и пожарная безопасность (ГОСТ 12.0.003-74 [23]). Пожарная безопасность осуществляется системой пожарной защиты и системой предотвращения пожара.

По взрыво- и пожароопасности все помещения, согласно техническому регламенту НПБ 105-95 [24], делятся на 5 категорий, в зависимости от применяемых на производстве веществ и их количества. Рассматриваемая учебная аудитория относится к пожароопасной категории В [25].

Основные причины возникновения пожаров:

- Нарушение правил пожарной безопасности;
- Перегрузка электросети;
- Неисправность прибора;
- Разряд молнии и неисправность молниеотвода.

Для того что бы избежать возникновения пожара необходимо проводить следующие профилактические работы, направленные на устранение возможных источников возникновения пожара:

- периодическая проверка проводки;
- отключение оборудования при покидании рабочего места;
- проведение инструктажа работников о пожаробезопасности.

Для предотвращения пожара в аудитории с ПЭВМ имеется:

- углекислотный огнетушитель типа ОУ-2 (данный тип огнетушителя подходит для помещений с электрооборудованием (ГОСТ Р 51057-01[26]);
- Пожарная сигнализация ДИП-ЗСУ (извещатель пожарный, дымовой оптико-электронный точечный).

#### **4.4. Экологическая безопасность**

В ходе работы, вредное воздействие возможно на литосферу. Это происходит из-за утилизации электронной техники: компьютеров, сканеров и т.п. Утилизация такого оборудования является достаточно сложной, так как такие они имеют сложную структуру. Непосредственная переработка большей части компонентов включает в себя их сортировку, последующую гомогенизацию и отправку для повторного использования, т.е. с предварительным помолом или переплавкой.

При рассмотрении влияния процесса утилизации персонального компьютера были выявлены особо вредные выбросы согласно ГОСТ Р 51768-2001 [27]. В случае выхода из строя компьютеров, они списываются и отправляются на специальный склад, который при необходимости принимает меры по утилизации списанной техники и комплектующих. В настоящее время в Томской области утилизацией занимаются две компании: городской полигон и ООО НПП «Экотом». Утилизацией опасных бытовых отходов занимаются компании: ООО «Торем», ООО «СибМеталлГрупп».

Процесс эксплуатации объекта не подразумевает никаких вредных аспектов, которые влияют на окружающую среду

#### **4.5. Безопасность в чрезвычайных ситуациях**

Одними из наиболее вероятных и разрушительных видов чрезвычайных ситуаций являются пожар.

Возникновение других ЧС при внедрении объекта нет.

Всякий работник при обнаружении пожара должен (ППБ 01-03 [28]):

- незамедлительно сообщить об этом в пожарную охрану;
- принять меры по эвакуации людей, каких-либо материальных ценностей согласно плану эвакуации;
- отключить электроэнергию, приступить к тушению пожара первичными средствами пожаротушения.

При возникновении пожара должна сработать система пожаротушения, передав на пункт пожарной станции сигнал о ЧС. В случае если система не сработала, то необходимо самостоятельно произвести вызов пожарной службы по телефону 101, сообщить точный адрес места возникновения ЧС и ожидать приезда специалистов.

Рабочее место располагается в 10 корпусе ТПУ 408 аудитория. На рисунке 35 представлен план эвакуации четвертого этажа 10 корпуса.

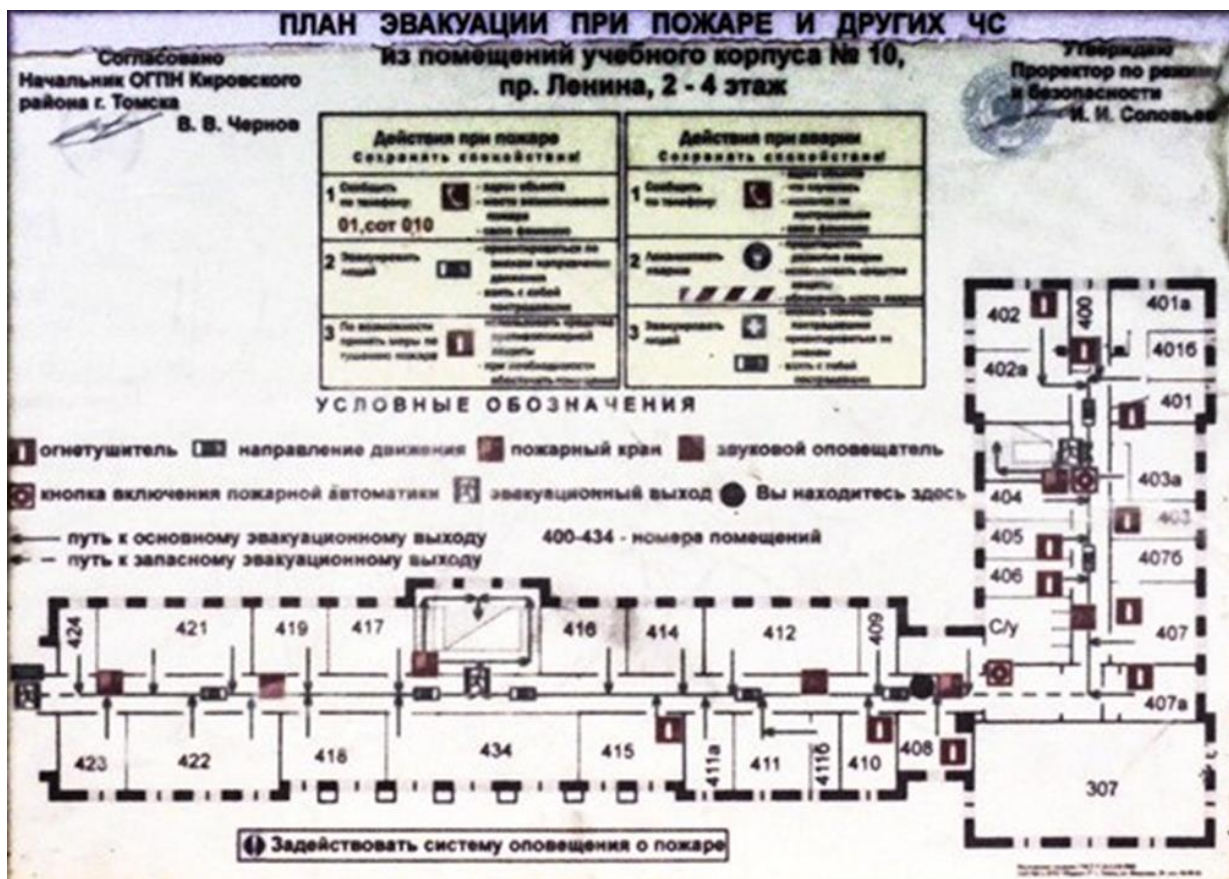


Рисунок 35. План эвакуации при чрезвычайных ситуациях

#### **4.6. Правовые и организационные вопросы обеспечения безопасности**

Согласно ТК РФ, N 197 -ФЗ работник имеет право на:

- рабочее место, соответствующее требованиям охраны труда; [29]
- обязательное социальное страхование от несчастных случаев на производстве и профессиональных заболеваний в соответствии с федеральным законом; [29]
- получение достоверной информации от работодателя, соответствующих государственных органов и общественных организаций об условиях и охране труда на рабочем месте, о существующем риске повреждения здоровья, а также о мерах по защите от воздействия вредных и (или) опасных производственных факторов; [29]
- отказ от выполнения работ в случае возникновения опасности для его жизни и здоровья вследствие нарушения требований охраны труда, за исключением случаев, предусмотренных федеральными законами, до устранения такой опасности; [29]
- обеспечение средствами индивидуальной и коллективной защиты в соответствии с требованиями охраны труда за счет средств работодателя; [29]
- обучение безопасным методам и приемам труда за счет средств работодателя; [29]
- личное участие или участие через своих представителей в рассмотрении вопросов, связанных с обеспечением безопасных условий труда на его рабочем месте, и в расследовании происшедшего с ним несчастного случая на производстве или профессионального заболевания; [29]
- внеочередной медицинский осмотр в соответствии с медицинскими рекомендациями с сохранением за ним места работы (должности) и среднего заработка во время прохождения указанного медицинского осмотра; [29]

- гарантии и компенсации, установленные в соответствии с настоящим Кодексом, коллективным договором, соглашением, локальным нормативным актом, трудовым договором, если он занят на работах с вредными и (или) опасными условиями труда. [29]

- повышенные или дополнительные гарантии и компенсации за работу на работах с вредными и (или) опасными условиями труда могут устанавливаться коллективным договором, локальным нормативным актом с учетом финансово-экономического положения работодателя. [29]

При планировке рабочего места необходимо учитывать следующее:

1. Рекомендуемый проход слева, справа и спереди от стола 500 мм. Слева от стола допускается проход 300 мм; [29]

2. Рабочие места с ПЭВМ при выполнении творческой работы, требующей значительного умственного напряжения или высокой концентрации внимания, рекомендуется изолировать друг от друга перегородками высотой 1,5-2,0 м. Экран видеомонитора должен находиться от глаз пользователя на расстоянии 600-700 мм, но не ближе 500 мм с учетом размеров алфавитно-цифровых знаков и символов. Дизайн ПЭВМ должен предусматривать окраску корпуса в спокойные мягкие тона с диффузным рассеиванием света. Корпус ПЭВМ, клавиатура и другие блоки и устройства ПЭВМ должны иметь матовую поверхность с коэффициентом отражения 0,4-0,6 и не иметь блестящих деталей, способных создавать блики. Конструкция рабочего стола должна обеспечивать оптимальное размещение на рабочей поверхности используемого оборудования с учетом его количества и конструктивных особенностей, характера выполняемой работы. При этом допускается использование рабочих столов различных конструкций, отвечающих современным требованиям эргономики; [29]

3. Конструкция рабочего стула (кресла) должна обеспечивать поддержание рациональной рабочей позы при работе на ПЭВМ позволять изменять позу с целью снижения статического напряжения мышц шейно-плечевой области и спины для предупреждения развития утомления. Тип

рабочего стула (кресла) следует выбирать с учетом роста пользователя, характера и продолжительности работы с ПЭВМ; [29]

4. Рабочий стул (кресло) должен быть подъемно-поворотным, регулируемым по высоте и углам наклона сиденья и спинки, а также расстоянию спинки от переднего края сиденья, при этом регулировка каждого параметра должна быть независимой, легко осуществляемой и иметь надежную фиксацию; [29]

5. Стул не может располагаться непосредственно на границе площади рабочего места. Рекомендуемое расстояние от спинки стула до границы должно быть не менее 300 мм. [29]

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА  
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И  
РЕСУРСОСБЕРЕЖЕНИЕ»**

Студенту:

Группа	ФИО
8В3А	Чайбар Артыш Алимович

Институт	ИК	Кафедра	ИСТ
Уровень образования	Бакалавриат	Направление/специальность	09.03.01 «Информатика и вычислительная техника»

**Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:**

<p><i>1. Организация и планирование работ:</i></p> <ul style="list-style-type: none"> <li>- Структура работы в рамках научного исследования;</li> <li>- Определение трудоемкости выполнения работ;</li> <li>- Разработка графика проведения научного исследования.</li> </ul> <p><i>Бюджет НИИ:</i></p> <ul style="list-style-type: none"> <li>- Расчет материальных затрат;</li> <li>- Расчет затрат на специальное оборудование для научных работ;</li> <li>- Расчет основной заработной платы исполнителей темы;</li> <li>- Расчет отчислений во внебюджетные фонды;</li> <li>- Расчет затрат на научные и производственные командировки;</li> <li>- Расчет контрагентных расходов;</li> <li>- Расчет накладных расходов;</li> <li>- Цена разработки НИИ.</li> </ul>	<p><i>1. В данном пункте проведена экспертная оценка основных технических характеристик конкурентных технических решений.</i></p> <p><i>2. Важным этапом реализации проекта является организация самого процесса и рациональное планирование работ для каждого исполнителя, принимающего участие в проекте. На данном этапе составляется полный перечень проводимых работ, определяются их исполнители и продолжительность, рассчитываются трудозатраты на выполнение проекта. Результатом планирования работ является таблица с полным перечнем, исполнителями и продолжительностью работ, а также таблица трудозатрат. Для наглядности представлен линейный график работ, построенный на основании приведенных таблиц.</i></p> <p><i>3. Целью данного пункта является расчет величины расходов на выполнение проекта. Определение общих затрат производится путем суммирования расходов по следующим статьям:</i></p> <ul style="list-style-type: none"> <li>• Материальные затраты НИИ;</li> </ul> <p><i>Затраты по данной статье отсутствуют.</i></p> <ul style="list-style-type: none"> <li>• Затраты на специальное оборудование для научных (экспериментальных) работ;</li> </ul> <p><i>Затраты по данной статье отсутствуют.</i></p> <ul style="list-style-type: none"> <li>• Основная заработная плата исполнителей темы;</li> </ul> <p><i>В данном пункте произведен расчет заработной платы как самого исполнителя, так и научного руководителя.</i></p> <ul style="list-style-type: none"> <li>• Дополнительная заработная плата исполнителей темы;</li> </ul> <p><i>Затраты по данной статье отсутствуют.</i></p> <ul style="list-style-type: none"> <li>• Отчисления во внебюджетные фонды (страховые отчисления);</li> </ul> <p><i>В этом пункте произведен расчет отчислений за единичный социальный налог, который в себя включает отчисления в пенсионный фонд, на социальное и медицинское страхование и</i></p>
---	---



	<p>составляет 30% от всей заработной платы</p> <ul style="list-style-type: none"> <li>• Затраты на научные и производственные командировки;</li> </ul> <p>Затраты по данной статье отсутствуют.</p> <ul style="list-style-type: none"> <li>• Контрагентные расходы;</li> </ul> <p>Затраты по данной статье отсутствуют.</p> <ul style="list-style-type: none"> <li>• Накладные расходы.</li> </ul> <p>В данном пункте отражены расходы на выполнение проекта, которые не учтены в предыдущих статьях, их следует принять равными 16% от суммы всех предыдущих расходов.</p>
--	---

<b>Дата выдачи задания для раздела по линейному графику</b>	
---	--

**Задание выдал консультант:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент каф. МЕН	Антонова И.С.	К.Э.Н.		

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8В3А	Чайбар Артыш Алимович		

## **5. ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСООБЪЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ**

### **5.1. Введение**

В данной работе рассматривается разработка создание клиент-серверного приложения для интерактивных табличек, которое формирует информацию для определенной аудитории и выводит полученную информацию на дисплей в режиме реального времени. Данные таблички устанавливаются возле каждой аудитории, на базе компьютеров Raspberry PI.

При создании табличек была определена следующая область применения:

- в офисных помещениях,
- для оперативного информирования о статусе должностных лиц в кабинетах,
- в учебных заведениях для информации и занятия, проводимом в аудитории,
- в медицинских учреждениях, которые предъявляют повышенные требования в оперативной информации о том или ином кабинете для посетителей,
- в социальных и сервисных службах, где требуется упорядочивание посещения кабинетов посетителями и др.

Целью данного раздела является анализ и описание проведенных научных исследований с финансово-экономической стороны, а также оценка полных денежных затрат, потраченных на реализацию проекта. В разделе дается экономическая оценка результатов внедрения данной работы, что в свою очередь, позволяет оценить целесообразность выполненной работы с экономической точки зрения.

## 5.2. Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения

### 5.2.1. Технология QuaD

Оценочная карта для сравнения конкурентных технических решений приведена в таблице 24:

Таблица 24 – Оценочная карта

Критерии оценки	Вес критерия	Баллы	Макс. балл	Отн. знач.	Ср.-взвеш. знач.
Показатели оценки качества разработки					
1. Скорость работы	0,2	100	100	1	0,2
2. Универсальность	0,2	95	100	0,95	0,19
3. Простота в эксплуатации	0,1	100	100	1	0,1
4. Ремонтпригодность	0,1	90	100	0,9	0,18
5. Качество веб-интерфейса	0,1	80	100	0,8	0,08
Показатели оценки коммерческого потенциала разработки					
6. Доступность	0,1	85	100	0,85	0,085
7. Конкурентоспособность продукта	0,05	75	100	0,75	0,0375
8. Перспективность на рынке	0,05	80	100	0,8	0,04
9. Финансовая эффективность научной разработки	0,05	85	100	0,85	0,0425
10. Цена	0,05	80	100	0,80	0,04
Итог:					0,995

Оценка качества и перспективности по технологии QuaD определяется по формуле:

$$P_{cp} = \sum V_i B_i, \quad (2)$$

где  $P_{cp}$  – средневзвешенное значение показателя качества и перспективности научной разработки;

$V_i$  – вес показателя (в долях единицы);

$B_i$  – балл  $i$ -го показателя.

Можно заметить, что интегральный показатель конкурентоспособности данной разработки составляет 0,995, что является достаточно благоприятным для продолжения разработки.

## 5.2.2. SWOT-анализ

Описание сильных и слабых сторон проекта, выявление возможностей и угроз. Результаты первого этапа представлены в таблице 25:

	<b>Сильные стороны научно-исследовательского проекта:</b> С1.Наличие открытой документации к интерфейсам. С2.Широкие возможности по масштабированию проекта. С3.Удобство в администрировании.	<b>Слабые стороны научно-исследовательского проекта:</b> Сл1.Зависимость от электро-энергии. Сл2.Отсутствие возможности доступной замены дисплея. Сл3.Постоянное наличие Интернет-соединения.
<b>Возможности</b> В1.Производство новых версий комплектующих. В2.Использование отечественных комплектующих. В3.Интеграция в любую структуру.	<b>В1С2:</b> Реализация новых функций и возможностей дисплеев. <b>В2С1:</b> Легко заменить используемую базу дисплеев на отечественные с наличием документации. <b>В3С3:</b> Позволит легко обучить персонал использовать данную систему, а так же администратора любой структуры.	<b>В1В2Сл1:</b> Использование внешних источников питания новых в версиях компьютеров, которые позволят работать системе при сбое электроэнергии. <b>В1Сл3:</b> Использование локального соединения, что не требует постоянного соединения с интернетом.
<b>Угрозы</b> У1.Развитая конкуренция технологий производства. У2.Введение дополнительных государственных требований к сертификации продукта У3.Повышение уровня брака комплектующих на рынке.	<b>У1С1:</b> Расширение использование дисплеев и системы управления ею. <b>У2С1:</b> Изменение функционала в соответствии с требованиями государства и соответственно документации. <b>У3С3:</b> Добавление функции оповещения неисправности какого-либо дисплея.	<b>У1Сл1:</b> Использовать внешние источники питания, позволит работу в автономном режиме. <b>У1Сл3:</b> Подключить к локальной сети, что позволит работать независимо от внешнего интернета. <b>У3Сл2:</b> Разработать корпус, который позволит легко заменять дисплеи.

Интерактивная матрица проекта представлена в табл. 26 и 27:

Таблица 26 – Интерактивная матрица проекта (сильные стороны)

	С1	С2	С3
В1	0	+	+
В2	+	+	+
В3	+	+	-

Таблица 27 – Интерактивная матрица проекта (слабые стороны)

	Сл1	Сл2	Сл3
B1	+	+	+
B2	+	+	-
B3	-	-	-

Таким образом, можно сделать вывод, что проект необходимо развивать, применяя наиболее новые и оптимизированные интерфейсы, что позволит создать наиболее конкурентоспособную разработку устройства.

### 5.3. Определение возможных альтернатив проведения научных исследований

В качестве морфологических характеристик в данной работе можно выделить язык программирования (подробней см. в основной части данной работы). Морфологическая матрица приведена в табл. 28:

Таблица 28 – Морфологическая матрица

Альтернативы	1	2	3	4
А. Язык программирования	PHP	Ruby	C#	Java

Выбор языка не имеет высокого значения в данном случае, так как все языки имеют примерно одинаковые возможности, но язык PHP (A1) предпочитают веб-программисты и был выбран из личного предпочтения.

### 5.4. Планирование научно-исследовательских работ

#### 5.4.1. Структура работ в рамках научного исследования

Перечень этапов и работ в рамках проведения научного исследования представлен в табл. 29:

Таблица 29 – Перечень этапов, работ и распределение исполнителей

№ этапа	Содержание работ	Исполнители	Загрузка исполнителей
1	Определение целей и задач	НР	НР – 100%
2	Составление и утверждение технического задания	НР, И	НР – 80% И – 20%
3	Разработка календарного плана	НР, И	НР – 80% И – 20%
4	Подбор и изучение материалов по теме	НР, И	НР – 10% И – 90%
5	Обсуждение литературы	НР, И	НР – 50% И – 50%
6	Проектирование информационной системы	НР, И	НР – 5% И – 95%
7	Программная реализация клиент-серверного приложения	И	И – 100%
8	Оформление расчетно-пояснительной записки	И	И – 100%
9	Оформление графического материала	И	И – 100%
10	Подведение итогов	НР, И	НР – 10% И – 90%

НР – научный руководитель, И – исполнитель.

#### 5.4.2. Определение трудоемкости выполнения работ

Оценим трудоемкость выполнения вышеозначенных работ. Для этого оценим минимальное и максимальное время выполнения каждой работы. Также произведем расчет ожидаемого значения трудоемкости по следующей формуле:

$$t_{ож,i} = \frac{(3t_{min,i} + 2t_{max,i})}{5}$$

где  $t_{ож,i}$  – ожидаемая трудоемкость выполнения  $i$ -ой работы, чел.-дн.;

$t_{min,i}$  – минимально возможная трудоемкость выполнения  $i$ -ой работы (оптимистическая оценка: в предположении наиболее благоприятного стечения обстоятельств), чел.-дн.;

$t_{max,i}$  – максимально возможная трудоемкость  $i$ -ой работы (пессимистическая оценка: в предположении наиболее неблагоприятного стечения обстоятельств), чел.-дн.

Следует заметить, что исполнитель задействован в каждой из перечисленных работ, а потому невозможно ускорение за счет параллельности их выполнения.

### 5.4.3. Разработка графика проведения научного исследования

Продолжительность этапа в календарных днях рассчитывается по формуле.

$$T_{ki} = T_{pi} \cdot k_{\text{кал}},$$

где  $T_{ki}$  – продолжительность выполнения  $i$ -й работы в календарных днях;

$T_{pi}$  – продолжительность выполнения  $i$ -й работы в рабочих днях;

$k_{\text{кал}}$  – коэффициент календарности, позволяющий перейти от длительности работ в рабочих днях к их аналогам в календарных днях и рассчитываемый по формуле (4).

$$k_{\text{кал}} = \frac{T_{\text{кал}}}{T_{\text{кал}} - T_{\text{вых}} - T_{\text{пр}}},$$

где  $T_{\text{кал}}$  – количество календарных дней в году ( $T_{\text{кал}} = 365$ );

$T_{\text{вых}}$  – количество выходных дней в году ( $T_{\text{вых}} = 52$ );

$T_{\text{пр}}$  – количество праздничных дней в году ( $T_{\text{пр}} = 10$ ).

Коэффициент календарности равен:

$$k_{\text{кал}} = \frac{365}{365 - 52 - 10} = 1,205$$

В таблице 30 представлена продолжительность этапов и их трудоемкость по исполнителям.

Таблица 30 – Трудозатраты на выполнение проекта

№ работ	Название работы	Трудоемкость работ, чел-дни			Длительность работ			
					$T_{pi}$		$T_{ki}$	
		$t_{min}$	$t_{max}$	$t_{ож}$	НР	И	НР	И
1	Определение целей и задач	2	3	2,4	2,4	0	2,89	0
2	Составление и утверждение технического задания	2	6	3,6	2,88	0,72	3,47	0,87
3	Разработка календарного плана	1	2	1,4	1,12	0,28	1,35	0,34
4	Подбор и изучение материалов по теме	4	12	7,2	0,72	6,48	0,87	7,81
5	Обсуждение литературы	3	6	4,2	2,1	2,1	2,53	2,53
6	Анализ существующих решений, выявление их преимуществ и недостатков	2	5	3,2	0,32	2,88	0,39	3,47
7	Проектирование информационной системы	5	15	9	0,45	8,55	0,54	10,30
8	Программная реализация клиент-серверного приложения	7	25	14,2	0	14,2	0	17,11
9	Оформление расчетно-пояснительной записки	5	7	5,8	0	5,8	0	6,99
10	Оформление графического материала	3	5	3,8	0	3,8	0	4,58



11	Подведение итогов	5	7	5,8	0,58	5,22	0,70	6,29
	<b>Итого:</b>			<b>60,6</b>	<b>10,57</b>	<b>50,03</b>	<b>12,73</b>	<b>60,27</b>

Зная величины трудоемкости этапов для каждого исполнителя  $T_{ki}$  можно построить календарный план-график. График строится для максимального по длительности исполнения работ в рамках научно-исследовательского проекта с разбивкой по месяцам и декадам (10 дней) за период времени дипломирования.

Таблица 31 – Календарный план-график проведения НИОКР по теме

№	Вид работ	Исполнители	$T_{ki}$ кал. дн	Продолжительность выполнения работ															
				март			апрель			май									
				1	2	3	1	2	3	1	2	3							
1	Определение целей и задач	НР	2,89																
2	Составление и утверждение технического задания	НР	3,47																
		И	0,87																
3	Разработка календарного плана	НР	1,35																
		И	0,34																
4	Подбор и изучение материалов по теме	НР	0,87																
		И	7,81																
5	Обсуждение литературы	НР	2,53																
		И	2,53																
6	Анализ существующих решений, выявление их преимуществ и недостатков	НР	0,39																
		И	3,47																
7	Проектирование информационной системы	НР	0,54																
		И	10,3																



монитор, дисплей для установки около аудитории и компьютеры Raspberry PI. Использовалось по одной единице каждого материального ресурса.

Расчет материальных затрат, использованных в данной работе, осуществляется по следующей формуле:

$$Z_M = (1 + k_T) * \sum_{i=1}^m C_{расхi} * N_{расхi}$$

где  $m$  – количество видов материальных ресурсов, потребляемых при выполнении научного исследования;

$N_{расхi}$  – количество материальных ресурсов  $i$ -го вида, планируемых к использованию при выполнении научного исследования;

$C_i$  – цена приобретения единицы  $i$ -го вида потребляемых материальных ресурсов;

$K_T$  – коэффициент, учитывающий транспортно-заготовительные расходы.

$$Z_M = (1 + 0.15) * (300 + 500 + 5430 + 30250 + 3210 + 2760) * 1 = 48817.5 \text{ руб.}$$

Таблица материальных затрат представлена в таблице 32:

Наименование	Единица измерения	Количество	Цена за ед., руб.	Затраты на материалы.
Системный блок	Руб/шт.	1	30250	30250
Мышь	Руб/шт.	1	300	300
Монитор	Руб/шт.	1	5430	5430
Клавиатура	Руб/шт.	1	500	500
Компьютер Raspberry PI	Руб/шт.	1	3210	3210
Дисплей	Руб/шт.	1	2760	2760
Итого:			42450	42450

### **5.5.2. Расчет затрат на специальное оборудование для научных (экспериментальных) работ**

В данную статью включают все затраты, связанные с приобретением специального оборудования (приборов, контрольно-измерительной аппаратуры, стенов, устройств и механизмов), необходимого для проведения работ по данной теме.

Затраты по данной статье расходов отсутствуют, так как никакое специальное оборудование не использовалось.

### **5.5.3. Основная заработная плата исполнителей темы**

В настоящую статью включается основная заработная плата научных и инженерно-технических работников, рабочих макетных мастерских и опытных производств, непосредственно участвующих в выполнении работ по данной теме. Величина расходов по заработной плате определяется исходя из трудоемкости выполняемых работ и действующей системы окладов и тарифных ставок. В состав основной заработной платы включается премия, выплачиваемая ежемесячно из фонда заработной платы в размере 20 – 30 % от тарифа или оклада.

Расчет основной заработной платы выполняется на основе трудоемкости выполнения каждого этапа и величины месячного оклада исполнителя.

Статья включает основную заработную плату работников, непосредственно занятых выполнением НИИ, (включая премии, доплаты) и дополнительную заработную плату:

$$ЗП_{зп} = З_{осн} + З_{доп}$$

где  $З_{осн}$  – основная заработная плата;

$З_{доп}$  – дополнительная заработная плата (12-20% от  $З_{осн}$ ).

Основная заработная плата руководителя (лаборанта, инженера) от предприятия (при наличии руководителя от предприятия) рассчитывается по следующей формуле:

$$Z_{\text{осн}} = Z_{\text{дн}} \cdot T_p$$

Где  $Z_{\text{осн}}$  – основная заработная плата одного работника;

$T_p$  – продолжительность работ, выполняемых научно-техническим работником, раб.дн.

$Z_{\text{дн}}$  – среднедневная заработная плата работника, руб.

Среднедневная заработная плата рассчитывается по формуле:

$$Z_{\text{дн}} = \frac{Z_m \cdot M}{F_d}$$

Где  $Z_m$  – месячный должностной оклад работника, руб;

$M$  – количество месяцев работы без отпуска в течение года:

При отпуске в 24 раб. дня  $M = 11,2$  месяца, 5-дневная неделя;

При отпуске в 48 раб. Дней  $M = 10,4$  месяца, 6-дневная неделя;

$F_d$  – действительный годовой фонд рабочего времени научно-технического персонала, раб.дн.

Месячный должностной оклад работника:

$$Z_m = Z_{\text{ТС}} \cdot (1 + k_{\text{пр}} + k_{\text{д}}) \cdot k_{\text{р}}$$

Где  $Z_{\text{ТС}}$  – заработная плата по тарифной ставке, руб.;

$k_{\text{пр}}$  – премиальный коэффициент, равный 0,3 (т.е. 30% от  $Z_{\text{ТС}}$ )

$k_{\text{р}}$  – районный коэффициент, равный 1,3 (для Томска).

Среднедневная тарифная заработная плата ( $Z_{\text{ТС}}$ ) для шестидневной рабочей недели рассчитывается по формуле (6).

$$Z_{\text{ТС}} = Z_m / 24,83$$

В таблице 33 представлены расчеты затрат на полную заработную плату:

Таблица 33 - Расчет заработной платы

Исполнитель	Оклад, руб./мес.	Среднедневная ставка, руб./раб.день	Затраты времени, раб.дни	Коэффициент	Фонд з/платы, руб.

Научный руководитель	18 888,88	698,2	12,73	1,699	15 100,85
Исполнитель	11 111,11	410,7	60,27	1,699	42 055,15
<b>Итого:</b>			73,00		<b>57 156</b>

#### 5.5.4. Отчисления во внебюджетные фонды

В данной статье расходов отражаются обязательные отчисления по установленным законодательством Российской Федерации нормам органам государственного социального страхования (ФСС), пенсионного фонда (ПФ) и медицинского страхования (ФФОМС) от затрат на оплату труда работников.

Затраты на единичный социальный налог (ЕСН), который включают в себя отчисления в пенсионный фонд, на социальное и медицинское страхование, составляют 30% от полной заработной платы по проекту. ЕСН рассчитывается по следующей формуле (7).

$$З_{внеб} = k_{внеб} * (З_{осн} + З_{доп})$$

Следовательно, ЕСН по данному проекту будет составлять:

Таблица 34 – Отчисления во внебюджетные фонды

Исполнитель	Основная заработная плата	Дополнительная заработная плата	Коэффициент	Отчисления во внебюджетные фонды
Научный руководитель	15 100,85	0	0,3	4 530,25
Исполнитель	42 055,15	0	0,3	12 616,54
<b>Итого:</b>				<b>17 146,79</b>

#### 5.5.5. Расчет затрат на научные и производственные командировки

Затраты на научные и производственные командировки исполнителей определяются в соответствии с планом выполнения темы и с учетом

действующих норм командировочных расходов различного вида и транспортных тарифов.

Затраты по данной статье расходов отсутствуют, т.к. все выполняется в Томске без выездов.

### **5.5.6. Контрагентные расходы**

Контрагентные расходы включают затраты, связанные с выполнением каких-либо работ по теме сторонними организациями (контрагентами, субподрядчиками).

Расчет величины этой группы расходов зависит от планируемого объема работ и определяется из условий договоров с контрагентами или субподрядчиками.

В работе затраты по данной статье расходов отсутствуют, т.к. все делается исполнителем и научным руководителем, то расходы по данной статье отсутствуют.

### **5.5.7. Накладные расходы**

Накладные расходы учитывают прочие затраты организации, не попавшие в предыдущие статьи расходов: лицензия, печать и ксерокопирование материалов исследования, оплата услуг связи, электроэнергии, почтовые и телеграфные расходы, размножение материалов и т.д. Их величина определяется по следующей формуле:

$$Z_{\text{накл}} = (\text{сумма статей } 1 \div 7) \cdot k_{\text{нр}},$$

где  $k_{\text{нр}}$  – коэффициент, учитывающий накладные расходы (16%).

Величина накладных расходов будет равна

$$Z_{\text{накл}} = (57156 + 17146,79 + 4000) \cdot 0,16 = 15888,44$$

### **5.5.8. Формирование бюджета затрат научно-исследовательского проекта**

Рассчитанная величина затрат научно-исследовательской работы является основой для формирования бюджета затрат проекта, который при формировании договора с заказчиком защищается научной организацией в качестве нижнего предела затрат на разработку научно-технической продукции.

Определение бюджета затрат на научно-исследовательский проект по каждому варианту исполнения приведен в таблице 7.

Таблица 35 – Расчет бюджета затрат НИИ

<b>Наименование статьи</b>	<b>Сумма, руб</b>	<b>Примечание</b>
1. Материальные затраты НИИ	42 450	Пункт 5.4.1
2. Затраты по основной заработной плате исполнителей темы	57 156	Пункт 5.4.3
3. Отчисления во внебюджетные фонды	17 146,79	Пункт 5.4.4
4. Накладные расходы	15 888,44	Пункт 5.4.7.
<b>Итого:</b>	132 641,2	



## ЗАКЛЮЧЕНИЕ

В ходе выполнения данной выпускной квалификационной работы по созданию интерактивных информационных табличек аудиторий на базе платформы Raspberry Pi была изучена структура информационной системы для работы клиент-серверного приложения, спецификации фреймворка bootstrap. Была глубже изучена работа с СУБД MySQL, работа с записями, связи между атрибутами разных таблиц. Кроме того, было написано клиент-серверное приложение, и было запущено на платформе Raspberry Pi. Данная система показала корректное отображение запрашиваемой информации, и были проверены все режимы работы. Были решены проблемы и вопросы, которые возникли в процессе разработки информационной системы. Таким образом, любой студент, приходящий в университет, сможет легко получить требуемую информацию о статусе аудитории/кабинета или статусе должностного лица. Так же, использование интерактивных табличек, позволит легко обновлять информацию для каждого кабинета/аудитории без затрат времени на создание новых дверных табличек.

## СПИСОК ПУБЛИКАЦИЙ АВТОРА

1. Чайбар А.А. Интерактивные дисплеи // Сборник трудов XIV Международной научно-практической конференции студентов, аспирантов и молодых ученых «Молодежь и современные информационные технологии». Томск, 7-11 ноября 2016 г. – 2017 – Томск: Изд-во ТПУ. – Т. 2. — С. 110 — 111.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. RASPBERRY PI 2 MODEL B [Электронный ресурс]. – URL: <https://www.raspberrypi.org/products/raspberry-pi-2-model-b/> (дата обращения: 10.03.2017).
2. Фреймворк – что это такое? Определение, значение, перевод [Электронный ресурс]. – URL: <https://что-ето-такое.ru/framework> (дата обращения: 11.03.2017).
3. Bootstrap [Электронный ресурс]. – URL: <https://mactersait.ru/news/terminologiya/bootstrap/> (дата обращения: 11.03.2017).
4. Какой лучший редактор HTML, PHP, CSS, JS кода? [Электронный ресурс]. – URL: <http://sitear.ru/material/luchshiy-redaktor-html-php-css-js-koda> (дата обращения: 11.03.2017).
5. Бретт М. PHP и MySQL. Исчерпывающее руководство. 2-е изд. – СПб.: Питер, 2017. – 544 с.: ил. – (Серия «Бестселлеры O'Reilly»).
6. Ajax Live Data Search using Jquery PHP MySql [Электронный ресурс]. – URL: <http://www.webslesson.info/2016/03/ajax-live-data-search-using-jquery-php-mysql.html> (дата обращения: 20.04.2017).
7. Регулярные выражения, пособие для новичков. Часть 1 [Электронный ресурс]. – URL: <https://habrahabr.ru/post/115825/> (дата обращения: 21.04.2017).
8. Регулярные выражения (шаблоны) [Электронный ресурс]. – URL: <http://www.php.su/articles/?cat=regex&page=008> (дата обращения: 21.04.2017).
9. Типы связей между таблицами (визуальные инструменты для баз данных) [Электронный ресурс]. – URL: [https://technet.microsoft.com/ru-ru/library/ms190651\(v=sql.105\).aspx](https://technet.microsoft.com/ru-ru/library/ms190651(v=sql.105).aspx) (дата обращения: 02.05.2017).

10. SQL ALTER TABLE Statement [Электронный ресурс]. – URL: [https://www.w3schools.com/sql/sql\\_alter.asp](https://www.w3schools.com/sql/sql_alter.asp) (дата обращения: 02.05.2017).
11. Components [Электронный ресурс]. – URL: <http://getbootstrap.com/components/> (дата обращения: 05.05.2017).
12. Самый простой пример реализации формы входа на сайт [Электронный ресурс]. – URL: <http://www.web-shpargalka.ru/primer-formi-vhoda-na-sait.php> (дата обращения: 03.03.2017).
13. Международный стандарт ICCSR26000:2011 «Социальная ответственность организации»
14. ГОСТ 12.0.003-74 ССБТ. Опасные и вредные производственные факторы. Классификация. - М.: Издательство стандартов, 2001. – 4 с.
15. СанПиН 2.2.2/2.4.1340-03. Гигиенические требования к персональным электронно-вычислительным машинам и организации работы. – М.: Информационно-издательский центр Минздрава России, 2003. – 54 с.;
16. СНиП 23-05-95. Естественное и искусственное освещение. – М.: Центр проектной продукции в строительстве, 2011. – 70 с.
17. ГОСТ 6825-91. Лампы люминесцентные трубчатые для общего освещения. – М.: Издательство стандартов, 1992. – 242 с.
18. Борьба с шумом на производстве: Справочник / Е.Я. Юдин, Л.А. Борисов; Под общ. ред. Е.Я. Юдина – М.: Машиностроение, 1985. – 400с.
19. ГОСТ 12.1.003-83. ССБТ. Общие требования безопасности. – М.: Издательство стандартов, 2002. – 13 с.
20. СНиП 23-03-2003. Защита от шума. – М.: Госстрой России, 2004. – 34 с.

21. СанПиН 2.2.4.548 – 96. Гигиенические требования к микроклимату производственных помещений. – М.: Информационно-издательский центр Минздрава России, 1997. – 20 с.
22. Безопасность жизнедеятельности. /Под ред. Н.А. Белова - М.: Знание, 2000 - 364с.
23. ГОСТ 12.0.003-74 ССБТ. Опасные и вредные производственные факторы. Классификация. - М.: Издательство стандартов, 2001. – 4 с.
24. НПБ 105-95. Определение категорий помещений и зданий по взрывопожарной и пожарной опасности. / Шебеко Ю.Н. – М.: ВНИИПО, 1998. – 119 с.
25. СП 12.13130.2009. Определение категорий помещений, зданий и наружных установок по взрывопожарной и пожарной опасности. – М.: Проспект, 2010. – 32 с.
26. ГОСТ Р 51057-01. Огнетушители переносные. Общие технические требования. Методы испытаний. – М.: Издательство стандартов, 2001. – 48 с.
27. ГОСТ Р 51768-2001. Ресурсосбережение. Обращение с отходами. Методика определения ртути в ртутьсодержащих отходах. Общие требования. – М.: Издательство стандартов, 2001. - 13 с.
28. ППБ 01-03. Правила пожарной безопасности в Российской Федерации. – М.: ФГУ ВНИИПО МЧС России, 2003. – 111 с.
29. Правовые и организационные вопросы обеспечения безопасности [Электронный ресурс]. – URL: [http://studbooks.net/1358777/menedzhment/pravovye\\_organizatsionny\\_e\\_voprosy\\_obespecheniya\\_bezopasnosti](http://studbooks.net/1358777/menedzhment/pravovye_organizatsionny_e_voprosy_obespecheniya_bezopasnosti) (дата обращения: 31.05.2017).

# ПРИЛОЖЕНИЕ А. ЛИСТИНГ КЛИЕНТ-СЕРВЕРНОГО ПРИЛОЖЕНИЯ ДЛЯ СТАТУСА КАБИНЕТА/АУДИТОРИИ (Справочное)

## Листинг скрипта формирования расписания (request.php)

```
<?php
header("Content-Type: text/html; charset=utf8");
require '/usr/www/art.fas.tpu.ru/scripts/conf.php';
require 'global_values.php';
//Получение номера аудитории, для получения его id и для формирования запроса на
расписание
$web_class = $_GET["id_class"];
$web_build = $_GET["id_build"];
$request_class = "SELECT * FROM class WHERE class = '$web_class' and build =
'$web_build'";
$query2 = mysql_query($request_class);
$table_class = mysql_fetch_assoc($query2);

$local_class = $table_class["class"];
$class_id = $table_class["id"];

//Установка часового пояса
date_default_timezone_set('Asia/Novosibirsk');

//Получаем текущий день недели и делаем запрос на ID дня недели
$cur_day = date("N");
$request_week = "SELECT * FROM week WHERE id_week='$cur_day'";
$query5 = mysql_query($request_week);
$table_week = mysql_fetch_assoc($query5);
$day = $table_week["id_week"];

$value1 = array();
$value2 = array();
//Определяем время начала и конца занятий
$start_end = "SELECT * FROM rasp WHERE id_week = '$cur_day' and id_class =
'$class_id'";
$quan = mysql_query($start_end);
while($need_time = mysql_fetch_assoc($quan)){
    $value1[] = $need_time["id_time"];
}
$value1 = array_unique($value1);
$value1 = array_values($value1);

$mx = max($value1) - 1;
$mn = min($value1) - 1;
$max = max($value1);
$min = min($value1);
$i = min($value1);

//Получаем расписание звонков и получаем требуемое время
$f = "SELECT * FROM time_lesson";
$x = mysql_query($f);
while($t = mysql_fetch_assoc($x)){
    $value2[] = $t;
}

$start_less = $value2[$mn]["start"];
```

```

$end_less = $value2[$mx]["end"];

//Получаем текущее время, чтобы получить соответствующие данные
$time_sys = date('G.i');
date('G.i.s');
if($time_sys < $start_less || $time_sys > $end_less){
    $end_lessons = true;
}
else{
    $time_sys = date('G.i');
    $request_timeLesson = "SELECT * FROM time_lesson WHERE start <= '$time_sys'
AND end >= '$time_sys'";
    $query6 = mysql_query($request_timeLesson);
    $table_timeLesson = mysql_fetch_assoc($query6);
    print_r ($table_timeLesson);
    echo $time = $table_timeLesson["id_time"];

    if(!($time == null)){
        //С таблицы с расписанием получаем ответ на запрос "получить текущее
раписание на сегодняшний день $id_day"
        $request_rasp = "SELECT * FROM rasp WHERE id_week = '$day' and id_time =
'$time' and id_class = '$class_id'"; //формирование запроса
        $query7 = mysql_query($request_rasp); //запрос к бд

        while($table_rasp = mysql_fetch_assoc($query7)){
            $value = $table_rasp["id_group"];

            $cur_user = $table_rasp["id_user"];
            $cur_less = $table_rasp["id_less"];
            $cur_type = $table_rasp["type_lesson"];
            $cur_order = $table_rasp["id_order"];

            $request_groups = "SELECT * FROM groups WHERE id_group =
'$value'"; //Запрос на получения списка групп
            $query3 = mysql_query($request_groups);
            $table_groups = mysql_fetch_assoc($query3);

            $g[] = $table_groups["num_group"]; //массив номеров групп
        }
        if($cur_type == 1)
            $cur_type = "ЛК";

        list($cur_surname, $cur_name, $cur_lastname) = get_user($cur_user);
        $cur_lesson = get_lesson($cur_less);

        //-----Первая пара-----
        --

        //если первое за день занятие, то предыдущее не выводим
        if($cur_order == $min){
            $min = $min + 1;
            $request_next = "SELECT * FROM rasp WHERE id_week = '$day' and
id_order = '$min'"; //формирование запроса
            $next = mysql_query($request_next); //запрос к бд

            while($next_row = mysql_fetch_assoc($next)){
                $value = $next_row["id_group"];

                $next_user = $next_row["id_user"];
                $next_less = $next_row["id_less"];
                $next_type = $next_row["type_lesson"];
            }
        }
    }
}

```

```

        $request_groups = "SELECT * FROM groups WHERE id_group =
'$value'"; //Запрос на получение списка групп
        $query3 = mysql_query($request_groups);
        $next_table = mysql_fetch_assoc($query3);
        $next_g[] = $next_table["num_group"]; //массив номеров групп
    }
    list($next_surname,      $next_name,      $next_lastname)      =
get_user($next_user);
    $next_lesson = get_lesson($next_less);
    $first = true;
}
else{
    $first = false;
}
}
//-----Последняя пара-----
--
    //если последнее за день занятие, то следующее не выводим
    if($cur_order == $max){
        $max = $max - 1;
        $request_last = "SELECT * FROM rasp WHERE id_week = '$day' and
id_order = '$max'"; //формирование запроса
        $last = mysql_query($request_last);
        //запрос к бд
        while($last_row = mysql_fetch_assoc($last)){
            $value = $last_row["id_group"];

            $last_user = $last_row["id_user"];
            $last_less = $last_row["id_less"];
            $last_type = $last_row["type_lesson"];

            $request_groups = "SELECT * FROM groups WHERE id_group =
'$value'"; //Запрос на получение списка групп
            $query3 = mysql_query($request_groups);
            $last_table = mysql_fetch_assoc($query3);
            $last_g[] = $last_table["num_group"]; //массив номеров групп
        }
        list($last_surname,      $last_name,      $last_lastname)      =
get_user($last_user);
        $last_lesson = get_lesson($last_less);
        $last = true;
    }
    else{
        $last = false;
    }
}
//-----Все остальные пары-----
--
    if(($cur_order > $min) && ($cur_order < $max)){
        $prev_order = $cur_order - 1;
        $r1 = "SELECT * FROM rasp WHERE id_week = '$day' and id_order =
'$prev_order' and id_class = '$class_id'"; //формирование запроса на предыдущую пару
        $prev_res = mysql_query($r1); //запрос к бд

        $next_order = $cur_order + 1;
        $r2 = "SELECT * FROM rasp WHERE id_week = '$day' and id_order =
'$next_order' and id_class = '$class_id'"; //формирование запроса на следующую пару
        $next_res = mysql_query($r2); //запрос к бд

        //Цикл получения данных о предыдущем занятии
        while($l_row = mysql_fetch_assoc($prev_res)){
            $value = $l_row["id_group"];

```



```

        $prev_user = $l_row["id_user"];
        $prev_less = $l_row["id_less"];
        $prev_type = $l_row["type_lesson"];
        if($prev_type == 1)
            $prev_type = "ЛК";
        $request_groups = "SELECT * FROM groups WHERE id_group =
'$value'"; //Запрос на получение списка групп предыдущего занятия
        $query3 = mysql_query($request_groups);
        $table_groups = mysql_fetch_assoc($query3);

        $prev_g[] = $table_groups["num_group"]; //массив номеров
групп предыдущего занятия
    }
    //Цикл получения данных о следующем занятии
    while($n_row = mysql_fetch_assoc($next_res)){
        $value = $n_row["id_group"];
        $follow_user = $n_row["id_user"];
        $follow_less = $n_row["id_less"];
        $follow_type = $n_row["type_lesson"];
        if($follow_type == 1)
            $follow_type = "ЛК";
        $request_groups = "SELECT * FROM groups WHERE id_group =
'$value'"; //Запрос на получение списка групп следующего занятия
        $query3 = mysql_query($request_groups);
        $table_groups = mysql_fetch_assoc($query3);

        $follow_g[] = $table_groups["num_group"]; //массив номеров
групп следующего занятия
    }

    list($prev_surname, $prev_name, $prev_lastname) =
get_user($prev_user); //Получение ФИО преподавателя предыдущего занятия
    list($follow_surname, $follow_name, $follow_lastname) =
get_user($follow_user); //Получение ФИО преподавателя следующего занятия
    $prev_lesson = get_lesson($prev_less); //Получение название
дисциплины предыдущего занятия
    $follow_lesson = get_lesson($follow_less); //Получение название
дисциплины следующего занятия
    }
}
else{
//Если перерыв, то выводим занятие, которое было до перерыва и занятие после
перерыва--
    $time_sys = date('G.i');
    $rest_time = "SELECT * FROM time_rest WHERE start_rest <= '$time_sys' AND
end_rest >= '$time_sys'";
    $q6 = mysql_query($rest_time);
    $rest_table = mysql_fetch_assoc($q6);
    $id_1 = $rest_table["id_rest"];
    $id_2 = $id_1 + 1;
    $rest_rasp1 = "SELECT * FROM rasp WHERE id_week = '$day' and id_order =
'$id_2' and id_class = '$class_id'"; //формирование запроса след занятия
    $q1 = mysql_query($rest_rasp1); //запрос к бд

    $rest_rasp2 = "SELECT * FROM rasp WHERE id_week = '$day' and id_order =
'$id_1' and id_class = '$class_id'"; //формирование запроса предыдущего занятия
    $q2 = mysql_query($rest_rasp2); //запрос к бд
    while($rest1 = mysql_fetch_assoc($q1)){
        $value = $rest1["id_group"];
        $rest_next_user = $rest1["id_user"];
        $rest_next_less = $rest1["id_less"];

```

```

        $request_groups = "SELECT * FROM groups WHERE id_group =
'$value'"; //Запрос на получения списка групп
        $query3 = mysql_query($request_groups);
        $table_groups = mysql_fetch_assoc($query3);
        $rest_next_group[] = $table_groups["num_group"]; //массив номеров
групп
    }
    while($rest2 = mysql_fetch_assoc($q2)){
        $value = $rest2["id_group"];
        $rest_prev_user = $rest2["id_user"];
        $rest_prev_less = $rest2["id_less"];
        $request_groups = "SELECT * FROM groups WHERE id_group =
'$value'"; //Запрос на получения списка групп
        $query3 = mysql_query($request_groups);
        $table_groups = mysql_fetch_assoc($query3);

        $rest_prev_group[] = $table_groups["num_group"]; //массив номеров
групп
    }
    list($rest_prev_surname, $rest_prev_name, $rest_prev_lastname) =
get_user($rest_prev_user); //Получение ФИО преподавателя предыдущего занятия
    list($rest_follow_surname, $rest_follow_name, $rest_follow_lastname) =
get_user($rest_next_user); //Получение ФИО преподавателя следующего занятия
    $rest_prev_lesson = get_lesson($rest_prev_less); //Получение название
дисциплины предыдущего занятия
    $rest_follow_lesson = get_lesson($rest_next_less); //Получение название
дисциплины следующего занятия
    $i++;
    $break = true;
}
}
?>

```

# ПРИЛОЖЕНИЕ Б. ЛИСТИНГ КЛИЕНТ-СЕРВЕРНОГО ПРИЛОЖЕНИЯ ДЛЯ СТАТУСА ДОЛЖНОСТНОГО ЛИЦА (Справочное)

## Настройки для подключения к базе данных (Conf.php).

```
<?php
require '/usr/www/art.fas.tpu.ru/config_files/app_config.php';
$mysqli = mysql_connect(DATABASE_HOST, DATABASE_USERNAME, DATABASE_PASSWORD);
$db = mysql_select_db(DATABASE_NAME, $mysqli);
mysql_query("SET character_set_results = 'utf8', character_set_client = 'utf8',
character_set_connection = 'utf8', character_set_database = 'utf8',
character_set_server = 'utf8'", $mysqli);
?>
```

## Листинг главной страницы (index.php).

```
<?php
header("Content-Type: text/html; charset=utf8");
require '/usr/www/art.fas.tpu.ru/scripts/function.php';
?>

<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>Вход</title>
  <link rel="stylesheet" href="style.css">

  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <!-- The above 3 meta tags *must* come first in the head; any other head content
  must come *after* these tags -->

  <!-- Bootstrap -->
  <link href="css/bootstrap.css" rel="stylesheet">
  <link href="css/style.css" rel="stylesheet">
  <script src="jquery/jquery-3.2.1.js"></script>

</head>
<body>
  <div class="container center-block" style="margin-top: 50px;">
    <div class="row">
      <div class="col-md-4 col-md-offset-4" style="margin-top:100px;">
        <h3 style="">Вход</h3>
        <form action="" method="post">
          <div class="form-group">
            <h4>Логин</h4>
            <input class="form-control" type="text" name="login"
placeholder="login">

            <h4>Пароль</h4>
            <input class="form-control" type="password" name="password"
placeholder="password" required>
            <span style="color: red;">
          </div>
        </form>
      </div>
    </div>
  </div>
</body>
</html>
</?php
```

```

        if(!empty($error_login))
            echo $error_login;
        ?>
    </span>
</div>
<button style="margin-bottom: 5px;" type="submit" class="btn btn-
success" name="send">Войти</button><br>

    </form>
</div>
</div>
</div>

<!-- Include all compiled plugins (below), or include individual files as needed
-->
<script src="js/bootstrap.js"></script>
</body>
</html>

```

### Листинг страницы личного кабинета пользователя (account.php).

```

<?php
    header("Content-Type: text/html; charset=utf8");
    require '/usr/www/art.fas.tpu.ru/scripts/function.php';
?>

<!DOCTYPE html>
<html>
<head>
    <meta charset="utf8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <!-- The above 3 meta tags *must* come first in the head; any other head content
    must come *after* these tags -->

    <!-- Bootstrap -->
    <link href="../../css/bootstrap.css" rel="stylesheet">
    <link href="../../css/style.css" rel="stylesheet">
    <link href="../../css/m_style.css" rel="stylesheet">
    <script src="../../jquery/jquery-3.2.1.js"></script>
    <meta charset="utf-8">
    <title>Личный кабинет</title>
</head>

<body>
    <div class="container center-block" style="margin-top: 100px;">
        <div class="row">
            <div class="col-md-4 col-sm-4 col-xs-4">

                </div>
                <?php
                    echo '<div class="col-md-4 col-sm-6 col-xs-4">
                        <div class="center-block" style="height: 100px; width:450px;">
                            <h3> 
                                '.$_SESSION["logged_user"]["surname"].'
                                '.$_SESSION["logged_user"]["name"].' <br> '.$_SESSION["logged_user"]["last_name"].'
                            </h3>
                            <p class="lead" style="margin-bottom: 0px;">
                                '.$_SESSION["logged_user"]["class"].' аудитория
                            </p>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>

```

```

        <div class="form-group">
            <form action="" method="post">
                <div class="style" align="left">
                    <p class="lead">Статус <br>
                    <select class="form-control" size="1" name="status">
                        <option disabled>Выберите статус</option>
                        <option value="Занят">Занят</option>
                        <option value="Отсутствую">Отсутствую</option>
                        <option value="На месте">На месте</option>
                        <option value="Перерыв">Перерыв</option>
                    </select></p>
                </div>

                <p class="lead">Дополнительная информация
                <textarea class="form-control" name="comment" cols="40"
rows="3"></textarea></p>

                <p><input class="btn btn-warning" type="submit" value="Выход"
name="exit">

                <input class="btn btn-success" type="submit" value="Обновить"
name="refresh"></p>
            </form>
        </div>
    </div>';
    ?>
</div>
</div>
<!-- Include all compiled plugins (below), or include individual files as needed -->
<script src="js/bootstrap.js"></script>
</body>
</html>

```

**Листинг страницы, отображающейся на дисплее. Отвечает за вывод статуса должностного лица (display.php).**

```

<?php
    header("Content-Type: text/html; charset=utf8");
    header('refresh: 3');
    require '/usr/www/art.fas.tpu.ru/scripts/function.php';
    $web_class = $_GET["id_c"];
    $web_build = $_GET["id_b"];

    $user = "SELECT * FROM users WHERE class = '$web_class' and build =
'$web_build'";
    $a = mysql_query($user);
    $row = mysql_fetch_assoc($a);

    if($row["active"] == 1)
        require '/usr/www/art.fas.tpu.ru/pages/display_on.php';
    else
        require '/usr/www/art.fas.tpu.ru/pages/display_off.php';
?>

```

**Листинг скрипта, отвечающий за вход в личный кабинет (Enter.php).**

```

<?php
    require '/usr/www/art.fas.tpu.ru/scripts/conf.php';
    session_start();

```

```

if(isset($_POST["send"])){

    $error_login = "";
    $error = false;

    //Массив данных пользователя
    $user = array();

    // Преобразование всего html-кода в специальные символы, чтобы не
    обрабатывались в бд
    $user = htmlspecialchars($_POST["login"]);
    $pass = htmlspecialchars($_POST["password"]);

    $pass = md5($pass);

    if(($user == "admin") && ($pass == md5("123456789"))){
        $_SESSION["admin"] = true;
        header('Location:admin.php');
    }
    else{
        // Исключаем пробелы
        $user = trim($user);

        //Подключение к бд и обработка данных
        $request = "SELECT * FROM users WHERE login='$user'";
        $query = mysql_query($request);
        $user_data = mysql_fetch_assoc($query);

        //Проверяем, верно ли введен логин и пароль
        if(($user_data["login"] != $user) || ($user_data["pass"] != $pass)){
            $error_login = "Введен неверный логин или пароль";
            $error = true;
        }
        if(!$error){
            $_SESSION["logged_user"] = $user_data;
            $var = $user_data["id"];
            $active = "UPDATE users SET active = '1' WHERE id = '$var'";
            $query1 = mysql_query($active);
            header('Location:pages/acount.php');
        }
    }
}
?>

```

**Код скрипта, отвечающий за обновление информации на дисплее**

**(Function.php).**

```

<?php
require '/usr/www/art.fas.tpu.ru/scripts/enter.php';

if(isset($_POST["exit"])){
    $val = $_SESSION["logged_user"]["login"];
    $add = "UPDATE users SET s_get = 'На месте' WHERE login = '$val'";
    $req1 = mysql_query($add);
    $remove = "UPDATE users SET c_get = '' WHERE login = '$val'";
    $req2 = mysql_query($remove);
    $active = "UPDATE users SET active = '0' WHERE login = '$val'";
    $req3 = mysql_query($active);
    unset($_SESSION["logged_user"]);
    mysql_close($mysqli);
    header("Location:../index.php");
}

```

```
}
else{
  if(isset($_SESSION["logged_user"]))
    if(isset($_POST["refresh"])){
      $status = trim($_POST["status"]);
      $comment = trim($_POST["comment"]);

      if($comment){
        $add = "UPDATE users SET s_get = '$status', c_get = '$comment'";
        $req = mysql_query($add);
      }else{
        $add = "UPDATE users SET s_get = '$status'";
        $req = mysql_query($add);
      }
    }
}
?>
```

# ПРИЛОЖЕНИЕ В. ЛИСТИНГ ПАНЕЛИ УПРАВЛЕНИЯ АДМИНИСТРАТОРА

(Справочное)

## Листинг страницы пользователя (user\_page.php).

```
<?php
$where = "";
header("Content-Type: text/html; charset=utf8");
require '/usr/www/art.fas.tpu.ru/admin/config_files/session.php';
require_once '/usr/www/art.fas.tpu.ru/scripts/conf.php';
$row="";
$row1="";
if(isset($_POST["exitA"])){
    unset($_SESSION["admin"]);
    header("Location:../../index.php");
}

if(isset($_GET["user_id"])){
    $id = $_GET["user_id"];
    $where = "WHERE id = ".$id;
}
if(isset($_POST["remove"])){
    $delete = "DELETE FROM users ".$where;
    $success = mysql_query($delete);
    if($success)
        header("Location:http://art.fas.aics.ru/admin.php");
}

//Получение id показываемого пользователя
$request_user = "SELECT * FROM users $where";
$result1 = mysql_query($request_user);
$row=mysql_fetch_assoc($result1);
$class = $row["class"];
$build = $row["build"];

//Получение id показываемой аудитории
$request_class = "SELECT * FROM class WHERE class = '$class' AND build =
'$build'";
$result2 = mysql_query($request_class);
$row1=mysql_fetch_assoc($result2);

mysql_close($mysqli);
?>

<html>
<head>
    <meta charset="utf-8"><?php
    echo '<title>'.$row["surname"].' '.$row["name"].'</title>'
    ?>
    <link rel="stylesheet" href="style.css">

    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <!-- The above 3 meta tags *must* come first in the head; any other head content
    must come *after* these tags -->
```



```

<!-- Bootstrap -->
<link href="../../css/bootstrap.css" rel="stylesheet">
<link href="../../css/style.css" rel="stylesheet">
<link href="https://api.tpu.ru/doc/json" rel=" "

<script type="text/javascript">
    function newwindow(form){
        var class_id = "<?php echo $row["class"]?>";
        var build_id = "<?php echo $row["build"]?>";

window.open(`http://art.fas.aics.ru/pages/display.php?id_c=${class_id}&id_b=${build_id}`);
    }
</script>
</head>

<body>
<nav class="navbar navbar-default navbar-inverse">
    <div class="container">
        <!-- Brand and toggle get grouped for better mobile display -->
        <div class="navbar-header">
            <button type="button" class="navbar-toggle collapsed" data-toggle="collapse"
data-target="#bs-example-navbar-collapse-1" aria-expanded="false">
                <span class="sr-only">Toggle navigation</span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
            </button>
            <span class="glyphicon glyphicon-cog" aria-hidden="true" style="font-size:
45px; padding-top: 2px; padding-right: 4px; color: white;"></span>
        </div>

        <!-- Collect the nav links, forms, and other content for toggling -->
        <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
            <ul class="nav navbar-nav">
                <li class="active"><a href="../../admin.php">Пользователи<span class="sr-
only">(current)</span></a></li>
                <li><a href="new_user.php">Добавить</a></li>
                <li><a href="remove.php">Поиск</a></li>
                <li><a href="admin_rasp.php">Расписание</a></li>
            </ul>
            <p class="navbar-text">Авторизован как админ</p>
            <ul class="nav navbar-nav navbar-right">
                <li><form action="" method="post" style="margin-bottom: 0px; padding-top:
7.5px;"><input type="submit" name="exitA" class="btn btn-inverse"
value="Выход"></form></li>
            </ul>
        </div><!-- /.navbar-collapse -->
    </div><!-- /.container-fluid -->
</nav>

<div class="row">
    <div class="col-md-4 col-md-offset-4">
        <?php
        echo
            '<div class="page-header" align="center" style="margin-bottom: 0 px;">
                <h1>'.$row["surname"].' '.$row["name"].' '.$row["last_name"].'<h1>
            </div>
            <div class="page-header">
                <h2 align="center">Информация пользователя</h2>
                <dl class="dl-horizontal" style="font-size: 14px;">

```

```

        <dt>Логин</dt>
        <dd>' . $row["login"] . '</dd>
        <dt>Рабочее место</dt>
        <dd>' . $row["build"] . ' корпус, ' . $row["class"] . ' аудитория</dd>
        <dt>Должность</dt>
        <dd>' . $row["info"] . '</dd>
    </dl>
    <form method="post" action="" style="margin-bottom: 0px;"><button
type="submit" class="btn btn-danger" name="remove" >Удалить
пользователя</button></form>
    </div>';
    echo
    '<div>
        <a href="edit.php?id=' . $row["id"] . '"><button type="submit" class="btn
btn-default" name="edit">Редактировать</button></a>
        <form action="#" method="get"><button type="submit" class="btn btn-
primary" name="look" onclick="newwindow(this)">Посмотреть дисплей</button></form>
    </div>';
    ?>
</div>
</div>
</body>
</html>

```

## Листинг страницы редактирования персональной информации пользователя (edit.php).

```

<?php
    $id = $_GET["id"];
    if($id)
        $where = "WHERE id = ".$id;

    if(isset($_POST["exitA"])){
        unset($_SESSION["admin"]);
        header("Location:../../index.php");
    }

    require '/usr/www/art.fas.tpu.ru/admin/config_files/update.php';

    //Получение id показываемого пользователя
    $mysql_request = "SELECT * FROM users $where";
    $result = mysql_query($mysql_request);
    $row=mysql_fetch_assoc($result);
    mysql_close($mysqlcli);
?>

<html>
<head>
    <meta charset="utf-8">
    <title>Add a new user</title>
    <link rel="stylesheet" href="style.css">

    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <!-- The above 3 meta tags *must* come first in the head; any other head content
    must come *after* these tags -->

    <!-- Bootstrap -->

```

```

    <link href="../../css/bootstrap.css" rel="stylesheet">
    <link href="../../css/style.css" rel="stylesheet">
    <link href="../../css/m_style.css" rel="stylesheet">
    <link href="https://api.tpu.ru/doc/json" rel=" " >
</head>

<body>
<nav class="navbar navbar-default navbar-inverse">
  <div class="container">
    <!-- Brand and toggle get grouped for better mobile display -->
    <div class="navbar-header">
      <button type="button" class="navbar-toggle collapsed" data-toggle="collapse"
data-target="#bs-example-navbar-collapse-1" aria-expanded="false">
        <span class="sr-only">Toggle navigation</span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </button>
      <span class="glyphicon glyphicon-cog" aria-hidden="true" style="font-size:
45px; padding-top: 2px; padding-right: 4px; color: white;"></span>
    </div>

    <!-- Collect the nav links, forms, and other content for toggling -->
    <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
      <ul class="nav navbar-nav">
        <li class="active"><a href="../../admin.php">Пользователи<span class="sr-
only">(current)</span></a></li>
        <li><a href="new_user.php">Добавить</a></li>
        <li><a href="remove.php">Поиск</a></li>
        <li><a href="admin_rasp.php">Расписание</a></li>
      </ul>
      <p class="navbar-text">Авторизован как админ</p>
      <ul class="nav navbar-nav navbar-right">
        <li><form action="" method="post" style="margin-bottom: 0px; padding-top:
7.5px;"><input type="submit" name="exitA" class="btn btn-inverse"
value="Выход"></form></li>
      </ul>
    </div><!-- /.navbar-collapse -->
  </div><!-- /.container-fluid -->
</nav>

  <!-- Columns start at 50% wide on mobile and bump up to 33.3% wide on desktop -->
<div class="row">
  <div class="col-md-4 col-md-offset-4">
    <?php
    echo
    '<div class="page-header">
      <h2>Обновление информации</h2>
    </div>
    <div>';
    if(empty($error))
      echo "";
    else{
      if($error == false)
        echo $success;
      else
        echo $text_error;
    }
    echo '<form action="" method="post">
      <div class="form-group">
        <h4>Новый логин</h4>

```

```

        <input class="form-control" type="text" name="newlogin"
value="'. $row["login"].' ">

        <h4>Новый пароль</h4>
        <input class="form-control" type="text" name="newPass">

        <h4>Новая фамилия пользователя</h4>
        <input class="form-control" type="text" name="newSurname"
value="'. $row["surname"].' ">

        <h4>Новое имя пользователя</h4>
        <input class="form-control" type="text" name="newName"
value="'. $row["name"].' ">

        <h4>Новое отчество пользователя</h4>
        <input class="form-control" type="text" name="newLastname"
value="'. $row["last_name"].' ">

        <h4>Аудитория</h4>
        <input class="form-control" type="text" name="newClass"
value="'. $row["class"].' ">

        <h4>Учебный корпус</h4>
        <input class="form-control" type="text" name="newBuild"
value="'. $row["build"].' ">

        <h4>Должность пользователя</h4>
        <input class="form-control" type="text" name="newInfo"
value="'. $row["info"].' ">
    </div>

    <button style="margin-bottom: 5px;" type="submit" class="btn btn-
primary" name="update">Обновить</button>
    <button style="margin-bottom: 5px;" class="btn btn-default"
name="cancel">Отмена</button>';
    echo '</form>
    </div>';
    ?>
</div>
</div>

</body>
</html>

```

### Скрипт обновления информации о пользователе (update.php).

```

<?php
require '/usr/www/art.fas.tpu.ru/admin/config_files/session.php';
require '/usr/www/art.fas.tpu.ru/scripts/conf.php';
global $error, $check, $text_error, $success;
$error = false;
if(isset($_POST["update"])){
    $new_login = trim($_POST["newlogin"]);
    $new_pass = trim($_POST["newPass"]);
    $new_name = trim($_POST["newName"]);
    $new_surname = trim($_POST["newSurname"]);
    $new_lastname = trim($_POST["newLastname"]);
    $new_class = trim($_POST["newClass"]);
    $new_build = trim($_POST["newBuild"]);
    $new_info = trim($_POST["newInfo"]);

    if(empty($new_login))

```

```

        $error = true;
    if(!$new_name)
        $error = true;
    if(!$new_surname)
        $error = true;
    if(!$new_lastname)
        $error = true;
    if(!$new_class)
        $error = true;
    if(!$new_build)
        $error = true;
    if(!$new_info)
        $error = true;
    if(!$new_pass){
        if(!$error){
            $request = "UPDATE users SET login='$new_login', name='$new_name',
surname = '$new_surname', last_name = '$new_lastname', class = '$new_class', info =
'$new_info', build = '$new_build' WHERE id = '$id'";
            $check = mysql_query($request);
            $success = '<div class="alert alert-success" role="alert">Информация
обновлена</div>';
        }
        else{
            $text_error = '<div class="alert alert-warning" role="alert">Не все
поля заполнены</div>';
        }
    }
    else{
        if(!$error){
            $new_pass = md5($new_pass);
            $request = "UPDATE users SET login='$new_login', pass='$new_pass',
name='$new_name', surname = '$new_surname', last_name = '$new_lastname', class =
'$new_class', info = '$new_info', build = '$new_build' WHERE id = '$id'";
            $check = mysql_query($request);
            $success = '<div class="alert alert-success" role="alert">Информация
обновлена</div>';
        }
        else{
            $text_error = '<div class="alert alert-warning" role="alert">Не все
поля заполнены</div>';
        }
    }
}
else{
    if(isset($_POST["cancel"])){
header('Location:http://art.fas.aics.ru/admin/user_page.php?user_id='.$id);
    }
}
?>

```

# ПРИЛОЖЕНИЕ Г. ЛИСТИНГ СКРИПТА СОЗДАНИЯ ПОЛЬЗОВАТЕЛЯ

(Справочное)

Листинг скрипта, отвечающего за добавления нового  
пользователя в систему (Create\_user.php).

```
<?php
require '/usr/www/art.fas.tpu.ru/admin/config_files/session.php';
require '/usr/www/art.fas.tpu.ru/scripts/conf.php';
$error = false;
$text_error = "";
$error_mas[] = array();
function mysql_insert($table, $inserts) {
    $values = array_map('mysql_real_escape_string', array_values($inserts));
    $keys = array_keys($inserts);
    return mysql_query('INSERT INTO `'.$table.'` (`.implode(``,`', $keys).`)
VALUES (`.implode(`\`,`\`,`', $values).'\`')');}
if(isset($_POST["create"])){
    $user_login = trim($_POST["login"]);
    $user_name = trim($_POST["name"]);
    $user_surname = trim($_POST["surname"]);
    $user_lastname = trim($_POST["last_name"]);
    $user_pass = trim($_POST["password"]);
    $user_class = trim($_POST["class"]);
    $user_info = trim($_POST["info"]);
    $user_build = trim($_POST["build"]);
    if(!$user_login)
        $error = true;
    if(!$user_name)
        $error = true;
    if(!$user_surname)
        $error = true;
    if(!$user_lastname)
        $error = true;
    if(!$user_pass)
        $error = true;
    if(!$user_class)
        $error = true;
    if(!$user_info)
        $error = true;
    if(!$error){
        $user_pass = md5($user_pass);
        mysql_insert('users', array(
            'login' => $user_login,
            'pass' => $user_pass,
            'name' => $user_name,
            'surname' => $user_surname,
            'last_name' => $user_lastname,
            'class' => $user_class,
            'info' => $user_info,
            'build' => $user_build
        ));
        $text_error = "Пользователь добавлен";}
    else{
        $text_error = "Ошибка добавления пользователя";}}
?>
```

# ПРИЛОЖЕНИЕ Д. ПОИСК ПОЛЬЗОВАТЕЛЕЙ В СИСТЕМЕ

(Справочное)

Листинг скрипта, отвечающего за поиск пользователей  
(search.php).

```
<?php
require '/usr/www/art.fas.tpu.ru/config_files/app_config.php';

$mysqli = mysql_connect(DATABASE_HOST, DATABASE_USERNAME, DATABASE_PASSWORD);

$db = mysql_select_db(DATABASE_NAME, $mysqli);
global $row;
mysql_query("SET character_set_results = 'utf8', character_set_client = 'utf8',
character_set_connection = 'utf8', character_set_database = 'utf8',
character_set_server = 'utf8'", $mysqli);
$output = '';
if(isset($_POST["query"]))
{
    $search = mysql_real_escape_string($_POST["query"]);
    $search = preg_replace("#[^\0-9a-ya-z]#iu","", $search);
    $query = "
        SELECT id, name, surname, last_name FROM users
        WHERE name LIKE '%".$search."%'
        OR surname LIKE '%".$search."%'
        OR last_name LIKE '%".$search."%'";

    $result = mysql_query($query);
    if(mysql_num_rows($result) > 0)
    {
        $output .= '
        <div class="table-responsive">
        <table class="table table-bordered">
        <tr>
            <th>Фамилия</th>
            <th>Имя</th>
            <th>Отчество</th>
            <th>Профиль</th>
        </tr>
        ';
        while($row = mysql_fetch_array($result))
        {
            $output .= '
            <tr>
                <td>'.$row["surname"].'</td>
                <td>'.$row["name"].'</td>
                <td>'.$row["last_name"].'</td>
                <td><a
href="http://art.fas.aics.ru/admin/user_page.php?user_id='.$row["id"].'"><input
type="submit" class="btn btn-primary" name="profile" value="Профиль"></td>
            </tr>';
        }
    }
    if(isset($_POST["remove"])){
        $delete = "DELETE FROM users WHERE id = '$id'";
        mysql_query($delete);
    }
}
echo $output;
```

```
}  
else  
{  
    echo 'Пользователи не найдены';  
}  
}
```

```
mysql_close($mysqli);
```

```
?>
```

### **Скрипт работы поиска, осуществляющий режим live search.**

```
<script>  
$(document).ready(function(){  
    load_data();  
    function load_data(query){  
        $.ajax({  
            url:"search.php",  
            method:"POST",  
            data:{query:query},  
            success:function(data){  
                $('#result').html(data);  
            }  
        });  
    }  
    $('#search_text').keyup(function(){  
        var search = $(this).val();  
        if(search != ''){  
            load_data(search);  
        }  
        else{  
            load_data();  
        }  
    });  
});  
</script>
```