|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **Министерство науки и высшего образования Российской Федерации**  Калужский филиал  федерального государственного бюджетного  образовательного учреждения высшего образования  ***«Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)»***  ***(КФ МГТУ им. Н.Э. Баумана)*** |

**ФАКУЛЬТЕТ** \_ ***ИУК «Информатика и управление»\_\_\_\_\_\_\_\_\_***

**КАФЕДРА** \_\_ ***ИУК5 «Системы обработки информации»***

**РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**к курсовой работе на тему:**

***Веб-приложение для автоматизации работы корпуса общественных наблюдателей Российского союза молодежи по Курской области.***

по дисциплине ***Базы данных***

Студент гр. ИУК5-51Б \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (Королев Е.С.)

(подпись) (Ф.И.О.)

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

(подпись) (Ф.И.О.)

Оценка руководителя \_\_\_\_\_ баллов \_\_\_\_\_\_\_\_\_\_\_

30-50 (дата)

Оценка защиты \_\_\_\_\_ баллов \_\_\_\_\_\_\_\_\_\_\_

30-50 (дата)

Оценка проекта \_\_\_\_\_ баллов \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(оценка по пятибалльной шкале)

Комиссия: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

(подпись) (Ф.И.О.)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

(подпись) (Ф.И.О.)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

(подпись) (Ф.И.О.)

Калуга, 2021

Калужский филиал   
федерального государственного бюджетного образовательного учреждения высшего образования

***«Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)»   
(КФ МГТУ им. Н.Э. Баумана)***

УТВЕРЖДАЮ

Заведующий кафедрой **\_\_ИУК5\_\_\_**

\_\_\_\_\_\_\_\_\_\_\_(Е.В. Вершинин)

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_г.

**ЗАДАНИЕ**

**на выполнение курсовой работы**

по дисциплине ***Базы данных***

Студент Королев Е.С. ИУК5-51Б \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(фамилия, инициалы, индекс группы)

Руководитель\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(фамилия, инициалы)

График выполнения проекта: 25% к\_4\_нед., 50% к\_7\_нед., 75% к\_10\_нед., 100% к\_14\_нед.

***1. Тема курсового проекта***

***Веб-приложение для автоматизации работы корпуса общественных наблюдателей Российского союза молодежи по Курской области.***

***2. Техническое задание***

*Разработать веб-приложение для автоматизации работы корпуса общественных наблюдателей Российского союза молодежи по Курской области.*

***3. Оформление курсового проекта***

3.1. Расчетно-пояснительная записка на\_\_\_\_\_\_\_ листах формата А4.

3.2. Перечень графического материала КП (плакаты, схемы, чертежи и т.п.)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата выдачи задания «\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2021г.

Руководитель курсового проекта \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

(подпись) (Ф.И.О.)

Задание получил\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/ «\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_2021г.

(подпись) (Ф.И.О.)

Примечание:

Задание оформляется в двух экземплярах: один выдается студенту, второй хранится на кафедре.

Оглавление

[1.Техническое задание 5](#_Toc89881351)

[1. Общие сведения 5](#_Toc89881352)

[1.1 Полное наименование системы и ее условное обозначение. 5](#_Toc89881353)

[1.2 Шифр темы или шифр (номер) договора. 5](#_Toc89881354)

[1.3 Перечень документов, на основании которых создается система. 5](#_Toc89881355)

[1.4 Наименование предприятий (объединений) разработчика и заказчика (пользователя) системы и их реквизиты. 5](#_Toc89881356)

[1.5 Плановые сроки начала и окончания работы по созданию системы. 5](#_Toc89881357)

[1.6 Сведения об источниках и порядке финансирования работ. 6](#_Toc89881358)

[2. Назначение и цели создания системы 6](#_Toc89881359)

[2.1 Назначение системы. 6](#_Toc89881360)

[2.2 Цели создания системы. 6](#_Toc89881361)

[3. Характеристика объекта автоматизации 6](#_Toc89881362)

[3.1 Краткие сведения об объекте автоматизации. 6](#_Toc89881363)

[4. Требования к системе 7](#_Toc89881364)

[4.1 Требования к системе в целом 7](#_Toc89881365)

[4.1.1 Требования к структуре и функционированию системы 7](#_Toc89881366)

[4.1.1.1 Требования к структуре системы. 7](#_Toc89881367)

[4.1.1.2 Требования к функционированию системы. 8](#_Toc89881368)

[4.1.2 Требования к надежности 8](#_Toc89881369)

[4.1.4 Требования к эргономике и технической эстетике 8](#_Toc89881370)

[4.2 Требования к функциям, выполняемым системой 9](#_Toc89881371)

[5. Состав и содержание работ по созданию системы 15](#_Toc89881372)

[6. Порядок контроля и приемки системы 16](#_Toc89881373)

[6.1 Состав, объем и методы испытаний системы и ее составных частей. 16](#_Toc89881374)

[6.2 Общие требования к приемке работ. 17](#_Toc89881375)

[7. Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие 17](#_Toc89881376)

[8. Требования к документированию 17](#_Toc89881377)

[9. Источники разработки 17](#_Toc89881378)

[2. Научно – исследовательская часть 18](#_Toc89881379)

[2.1 Постановка задачи проектирования 18](#_Toc89881380)

[2.2. Описание предметной области 18](#_Toc89881381)

[2.3 Анализ аналогов. 21](#_Toc89881382)

[2.4 Перечень задач, подлежащих решению в процессе разработки. 21](#_Toc89881383)

[2.5 Обоснование выбора инструментов и платформы для разработки. 21](#_Toc89881384)

[Особенности Visual Studio Code 21](#_Toc89881385)

[3. Проектно- конструкторская часть 27](#_Toc89881386)

[3.1 Логическая схема базы данных 27](#_Toc89881387)

[3.2 Физическая схема базы данных 27](#_Toc89881388)

[3.3 Описание сущностей базы данных 28](#_Toc89881389)

[3.4 Sequence 33](#_Toc89881390)

[3.5 Хранимые процедуры и функции 34](#_Toc89881391)

[3.6 Триггеры 35](#_Toc89881392)

[3.7 Нормализация базы данных 36](#_Toc89881393)

[3.8 Обеспечение целостности базы данных 37](#_Toc89881394)

[4. ПРОЕКТНО-ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ 38](#_Toc89881395)

[4.1 Процедуры автоматизации для обеспечения отказоустойчивости. 38](#_Toc89881396)

[4.2 Порядок развёртывания системы. 38](#_Toc89881397)

[4.3 Разработка руководства пользователя и руководства администратора. 39](#_Toc89881398)

[4.4 Тестирование и отладка рабочей программы. 48](#_Toc89881399)

[ЗАКЛЮЧЕНИЕ 52](#_Toc89881400)

[ЛИТЕРАТУРА 53](#_Toc89881401)

### 1.Техническое задание

### 1. Общие сведения

### 1.1 Полное наименование системы и ее условное обозначение.

Полное наименование системы: Региональный портал корпуса общественных наблюдателей Российского союза молодежи по Курской области. Условное обозначение: «Система».

### 1.2 Шифр темы или шифр (номер) договора.

Курсовая работа на тему «Разработка веб-приложения для обеспечения работы корпуса общественных наблюдателей», от 01.09.2021 г.

### 1.3 Перечень документов, на основании которых создается система.

Методические указания к выполнению курсовой работы по дисциплинам «Базы данных» и «Архитектура автоматизированных систем обработки информации и управления».

### 1.4 Наименование предприятий (объединений) разработчика и заказчика (пользователя) системы и их реквизиты.

Заказчик: корпус общественных наблюдателей Российского союза молодежи Курской области в лице заместителя регионального координатора Курской области Арцыбашева Данила Александровича.

Разработчик: студент группы ИУК5-51Б Королев Евгений Сергеевич.

Телефон: +7(910)868-64-42.

### 1.5 Плановые сроки начала и окончания работы по созданию системы.

Начало разработки – 01.09.2021 г.

Окончание разработки – 12.12.2021 г.

### 1.6 Сведения об источниках и порядке финансирования работ.

Источник финансирования не имеется.

### 2. Назначение и цели создания системы

### 2.1 Назначение системы.

Система предназначена для автоматизации процесса обучения общественных наблюдателей, автоматического создания документации для оформления пропусков на пункты проведения экзаменов, заполнения отчета о прохождении экзамена.

### 2.2 Цели создания системы.

Целями создания Системы являются:

1. Разработка программного обеспечения, автоматизирующего работу корпуса общественных наблюдателей;
2. Централизация обучения наблюдателей;
3. Автоматизация заполнения требуемой документации;
4. Централизация заполнения, автоматизация сбора и хранения отчетов о наблюдении в ППЭ;
5. Использование на практике навыков по разработке и реализации прикладного веб-приложения с использованием базы данных;

### 3. Характеристика объекта автоматизации

### 3.1 Краткие сведения об объекте автоматизации.

Объектом автоматизации является работа корпуса общественных наблюдателей Курской области (далее - корпус). Корпус ведет работу по обучению федеральных общественных наблюдателей (далее -наблюдатели), отправляемых на проверку пунктов проведения экзамена (далее - ППЭ), и их аттестации. В процессе обучения наблюдатели проходят 12 занятий, одно из которых аттестационное. Проводятся занятия одним из руководителей корпуса с возможным использованием электронного обучения.

По результатам прохождения обучения на успешно окончивших его наблюдателей оформляется пакет документов для допуска в ППЭ в день проведения экзамена. На основании этого пакета документа наблюдателям выдается пропуск.

В день проведения экзамена наблюдатель, закрепленный за определенным ППЭ, прибывает в ППЭ с 8:00 до 9:00 и начинает наблюдение за ППЭ. С 8:00 до 9:00 отчитывается о: времени прибытия в ППЭ, времени начала наблюдения, ФИО руководителя ППЭ, количестве аудиторий, ФИО членов ГЭК, ФИО общественных наблюдателей, ФИО федеральных общественных наблюдателей. С 9:00 до 10:15 отчитывается о нарушениях на входе в ППЭ. С 10:00 до 10:30 отчитывается о времени начала экзамена. С 10:00 до 17:00 отчитывается о: нарушениях во время проведения экзамена, времени окончания экзамена. С 12:00 до 18:30 отчитывается о: прочих замечаниях, времени окончания контроля.

После сдачи отчета о времени окончания контроля в ППЭ, наблюдатель покидает ППЭ.

### 4. Требования к системе

### 4.1 Требования к системе в целом

Разработка Системы должна проводиться с учетом следующих требований:

* Использование СУБД Oracle 10XE.
* Язык программирования для реализации серверной части приложения PHP.
* Язык программирования JavaScript, язык разметки HTML и таблицы стилей CSS.

### 4.1.1 Требования к структуре и функционированию системы

### 4.1.1.1 Требования к структуре системы.

Структура Системы должна включать в себя следующие составляющие:

* Графический интерфейс пользователя, необходимый для взаимодействия с системой.
* Сервер веб-приложения, предназначенный для соединения приложения с базой данных, маршрутизации ссылок страниц, хранения необходимых файлов.
* Сервер базы данных Oracle, выполняющий обслуживание и управление базой данных и отвечающий за целостность и сохранность данных, а также обеспечивающий операции ввода-вывода при доступе клиента к информации.

### 4.1.1.2 Требования к функционированию системы.

Система должна функционировать после запуска веб-сервера и сервера базы данных, после чего пользователь, перейдя по ссылке маршрутизации на главную страницу перейдет к информационной странице корпуса общественных наблюдателей Курской области.

### 4.1.2 Требования к надежности

Программа должна обеспечивать корректную обработку исключительных ситуаций, вызванных неверными действиями пользователей, неверным форматом или недопустимыми значениями входных данных. В указанных случаях система должна выдавать пользователю соответствующие аварийные сообщения, после чего возвращаться в рабочее состояние, предшествовавшее неверным действиям или некорректному вводу данных.

### 4.1.3 Требования к эргономике и технической эстетике

Взаимодействие пользователей с программным комплексом должно осуществляться посредством визуального графического интерфейса. Ввод-вывод данных, прием управляющих команд и отображение результатов их исполнения должны выполняться в интерактивном режиме, в реальном времени.

Интерфейс управления должен быть реализован с помощью набора экранных меню, кнопок, значков и т.п. элементов. Клавиатурный ввод должен использоваться при заполнении / редактировании текстовых полей экранной формы.

Пользовательский интерфейс разделен на две части: пользовательский интерфейс общественного наблюдателя (пользователя) и интерфейс руководителя и заместителя руководителя корпуса общественных наблюдателей (администратора).

### 4.2 Требования к функциям, выполняемым системой

**4.2.1 Авторизация**

Данная функция должна срабатывать непосредственно после нажатия кнопки «ВОЙТИ» на странице авторизации. Авторизация осуществляется по паре: логин, пароль.

**Требования к реализации**

Функция должна перенаправить пользователя на страницу с требуемым набором возможностей в зависимости от типа пользователя, корректно обработать исключительную ситуацию, при нахождении в БД несколько подходящих ученых записей, обработать ситуацию, если совпадений в БД не найдено.

**4.2.2 Функция заполнения анкеты пользователя**

Функция доступна всем типам пользователей на странице личного кабинета. По нажатию пользователем на кнопку «Анкетные данные» должна открываться соответствующая форма, после заполнения которой и нажатия на кнопку «Сохранить» введенные данные запишутся в соответствующие столбцы в таблице БД.

Список анкетных данных:

* Пол
* День рождения
* Место рождения
* Паспорт
* Кем выдан
* Дата выдачи
* Код подразделения
* Место учебы(работы)
* Должность
* Курс
* Группа
* Факультет
* Форма обучения
* Адрес регистрации
  + Область
  + Район
  + Населенный пункт
  + Улица
  + Дом
  + Квартира
* Адрес проживания
  + Область
  + Район
  + Населенный пункт
  + Улица
  + Дом
  + Квартира
* Год вступления в КОН
* Согласие на ОПД

**Требования к реализации**

Функция должна корректно записывать данные о пользователе в БД и обрабатывать исключительные ситуации, связанные с неверными действиями пользователя или форматом введенных данных.

**4.2.3 Функция загрузки личных документов.**

Функция доступна всем типам пользователей на странице личного кабинета. По нажатию пользователем на кнопку «Личные документы» должна открываться список загруженных на портал документов текущим пользователем и меню управления документами, в котором можно загрузить файл с компьютера при нажатии кнопки «Загрузить», удалить файл при выборе документа в «чекбоксе» и нажатии кнопки «Удалить», установка выбранного файла в качестве фотографии учетной записи по нажатию кнопки «использовать, как личное фото», после заполнения которой пользователь должен войти в личный кабинет.

**Требования к реализации**

Функция должна корректно обрабатывать исключительные ситуации, связанные с неверными действиями пользователя или форматом выбранных файлов. Файл сохранить в файловой системе веб сервера, расположение файла записать в БД.

**4.2.4 Функция просмотра текущих проверок.**

Функция доступна администраторам на странице текущие проверки. Отображает план проверки ППЭ с прикрепленными к ним общественными наблюдателями. После заполнения наблюдателем отчета, отображается ссылка на отчет. На странице должна находиться кнопка перенаправляющая на Google форму. Для каждого наблюдателя реализована возможность оценить его работу.

**Требования к реализации**

Функция должна корректно обрабатывать исключительные ситуации, связанные с неверными действиями пользователя. Выгрузка данных из БД и их изменение должно происходить корректно.

**4.2.5 Функция просмотра общих документов.**

Функция доступна всем типам пользователей на странице документы, выводить название документа и название файла. Отображает весь список загруженных документов, доступных для просмотра/ загрузки. Для администраторов доступно меню загрузки и удаления документов. При загрузке открывается форма, в которой можно указать название файла, которое будет в дальнейшем отображаться в списке файлов.

**Требования к реализации**

Функция должна корректно обрабатывать исключительные ситуации, связанные с неверными действиями пользователя. Выгрузка данных из БД и их изменение должно происходить корректно.

**4.2.6 Функция заполнения журнала.**

Функция доступна администраторам на странице занятия. Отображает журнал со всеми пользователями для выставления оценок (+, 1, 2, 3, 4, 5, нб(отр)) по 12 занятиям и экзамену. При нажатии кнопки «Ведомость» формируется ведомость в формате .xlsx и загружается на устройство.

**Требования к реализации**

Функция должна корректно обрабатывать исключительные ситуации, связанные с неверными действиями пользователя. Выгрузка данных из БД и их изменение должно происходить корректно.

**4.2.7 Функция создания занятий.**

Функция доступна администраторам на странице занятия. Отображается список из 12 занятий и экзамена, доступных для редактирования. Возможно заполнение столбцов: Дата, Время, № темы, Тема занятия, Материал, Руководитель.

**Требования к реализации**

Функция должна корректно обрабатывать исключительные ситуации, связанные с неверными действиями пользователя. Выгрузка данных из БД и их изменение должно происходить корректно.

**4.2.7 Функция создания плана контроля.**

Функция доступна администраторам на странице план контроля. Отображается список проверок ППЭ, состоящий из даты проверки, наименования ППЭ, проходящего экзамена, назначенного наблюдателя, проверяющего (администратора), ссылки на отчет, оценки проверки. Возможность выбора временного промежутка для загрузки плана. При нажатии кнопки «Отчет» происходит формирование отчета в формате .xlsx и загрузка на устройство. При нажатии на кнопку «Добавить» добавляется пустая строка в план, которую можно редактировать.

**Требования к реализации**

Функция должна корректно обрабатывать исключительные ситуации, связанные с неверными действиями пользователя. Выгрузка данных из БД и их изменение должно происходить корректно.

**4.2.8 Функция управления сотрудниками**

Функция доступна администраторам на странице сотрудники. Отображается список всех сотрудников, в таблице отображаются поля: №, столбец с кнопками управления, Фото, ФИО, Логин, Ранг, Телефон, Почта, Статус(активен/ блокирован).

При нажатии кнопки редактирования открывается форма редактирования учетной записи, в которой можно изменить ФИО, Логин, Телефон пользователя, Почту, Ранг пользователя (ФОН; ЗРКОН; РКОН; ФЭ; ФЭ.РКОН; ФЭ,ЗРКОН), Статус, Пароль. При нажатии кнопки «Сохранить» в форме данные сохраняются в БД. При нажатии кнопки «Удалить» учетная запись пользователя удаляется, войти на портал по ее логину и паролю становится невозможно.

При нажатии кнопки загрузки пакета документов пользователя, формируется пакет документов в формате .xlsx и загружается на устройство.

При выборе пользователя по нажатию на кнопку «Удалить» можно удалить выбранного пользователя, либо список выбранных пользователей.

При нажатии кнопки «Создать» создается новая учетная запись пользователя.

Реализована возможность поиска сотрудника по различным полям.

**Требования к реализации**

Функция должна корректно обрабатывать исключительные ситуации, связанные с неверными действиями пользователя. Выгрузка данных из БД и их изменение должно происходить корректно.

**4.2.9 Функция загрузки отчета по пользователям.**

Функция доступна администраторам на странице сотрудники. При нажатии кнопки «Отчет» происходит формирование отчета в формате .xlsx и загрузка на устройство.

**Требования к реализации**

Функция должна корректно обрабатывать исключительные ситуации, связанные с неверными действиями пользователя. Выгрузка данных из БД должна происходить корректно.

**4.2.10 Функция создания и просмотра ППЭ.**

Функция доступна администраторам на странице пункты ППЭ. Отображается список всех сохраненный в системе ППЭ. В таблице отображаются поля: №, Код ППЭ, Наименование образовательного учреждения, адрес ППЭ. Все поля кроме номера доступны для редактирования.

При нажатии кнопки «Добавить» добавляется пустая запись, которую можно редактировать.

При нажатии кнопки «Отчет» происходит формирование отчета в формате .xlsx и загрузка на устройство.

**Требования к реализации**

Функция должна корректно обрабатывать исключительные ситуации, связанные с неверными действиями пользователя. Выгрузка данных из БД и загрузка изменений должны происходить корректно.

**4.2.11 Функция создания и просмотра экзаменов.**

Функция доступна администраторам на странице экзамены. Отображается список всех сохраненный в системе экзаменов. В таблице отображаются поля: №, Код экзамена, Тип экзамена, Экзамен. Все поля кроме номера доступны для редактирования.

При нажатии кнопки «Добавить» добавляется пустая запись, которую можно редактировать.

При нажатии кнопки «Отчет» происходит формирование отчета в формате .xlsx и загрузка на устройство.

**Требования к реализации**

Функция должна корректно обрабатывать исключительные ситуации, связанные с неверными действиями пользователя. Выгрузка данных из БД и загрузка изменений должны происходить корректно.

**4.2.12 Функция создания отчета о наблюдении в ППЭ.**

Функция доступна пользователям на странице текущий контроль. При нажатии кнопки «Отчет» открывается окно бланка отчета, который пользователь заполняет. Определенные поля доступны для заполнения в определенное время, список полей и временных рамок:

С 8:00 до 9:00:

* время прибытия в ППЭ;
* время начала наблюдения;
* ФИО руководителя ППЭ;
* количество аудиторий;
* ФИО членов ГЭК;
* ФИО общественных наблюдателей;
* ФИО федеральных общественных наблюдателей.

С 9:00 до 10:15:

* нарушения на входе в ППЭ.

С 10:00 до 10:30:

* время начала экзамена.

С 10:00 до 17:00:

* нарушениях во время проведения экзамена;
* время окончания экзамена.

С 12:00 до 18:30:

* прочие замечания;
* время окончания контроля.

**Требования к реализации**

Функция должна корректно обрабатывать исключительные ситуации, связанные с неверными действиями пользователя. Выгрузка данных из БД и сохранение данных должны происходить корректно.

### 5. Состав и содержание работ по созданию системы

Состав и содержание работ по созданию чата указаны в таблице 1.1.

Таблица 1.1. Календарный план работ по созданию Системы.

|  |  |  |
| --- | --- | --- |
| Наименование стадий и этапов создания системы | Сроки выполнения работ | Результаты работ |
| 1.Формироване требований к АС и разработка технического задания.  1.1. Выбор и согласование темы с научным руководителем.  1.2. Проведение исследования по выбранной теме.  1.3. Обзор существующих программных продуктов.  1.4. Разработка технического задания. | 01.09.21-10.09.21 | Составленное техническое задание по стандарту ГОСТ 34.602-89, ГОСТ 34.601-90. |
| 2.Реализация программного обеспечения.  2.1. Разработка и организация алгоритмов функционирования приложения и интерфейса взаимодействия пользователя с интерфейсом.  2.2 Выполнение и оформление проектно-конструкторской части работы. | 11.09.21-22.10.21 | Разработанное приложение. Оформленная проектно-конструкторская часть согласно ГОСТ. |
| 3.Тестирование приложения и разработка дополнительной документации  3.1. Тестирование и отладка программного обеспечения.  3.2. Разработка руководства пользователя или программиста.  3.3. Написание проектно-технологической части и расчетно-пояснительной записки. | 23.10.21-22.11.21 | Разработанное руководство пользователя или программиста. |
| 4. Завершение оформления документации и защита курсовой работы  4.1. Завершение оформления документации согласно требованиям ГОСТ.  4.2. Подготовка доклада и защита. | 23.11.21-12.12.21 | Работоспособное приложение, отвечающее требованиям настоящего технического задания.  Оформленная необходимая документация к проекту |

### 6. Порядок контроля и приемки системы

### 6.1 Состав, объем и методы испытаний системы и ее составных частей.

Первая версия Системы должна пройти предварительные испытания, состоящие из отладки и дымового тестирования.

По итогам предварительных испытаний в Систему должны быть внесены исправления, учитывающие замечания, полученные в ходе предварительных испытаний.

Для проверки результата внесенных изменений должны быть проведены повторные испытания по ранее разработанной программе.

Повторные испытания включают в себя проверку работы функций Системы. Основной целью является проверка реализации Системы на соответствие требованиям настоящего «Технического задания».

### 6.2 Общие требования к приемке работ.

В процессе приемки работ должна быть осуществлена проверка Системы на соответствие требованиям настоящего «Технического задания». По результатам испытаний возможны доработки и исправления.

### 7. Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие

Для функционирования приложения достаточно запустить веб сервер и сервер СУБД, и перейти в браузере по ссылке маршрутизации на главную страницу.

### 8. Требования к документированию

По окончанию работы предъявлена расчетно-пояснительная записка в состав которой входят:

* техническое задание;
* научно-исследовательская часть;
* проектно-конструкторская часть;
* проектно-технологическая часть.

Также должна быть предоставлена графическая часть работы, выполненная формате А1 на 2 листах, в которую входят:

демонстрационные чертежи;

алгоритмические схемы.

### 9. Источники разработки

ГОСТ 34.602-89 [1]

ГОСТ 34.601.90 [2]

# 2. Научно – исследовательская часть

## 2.1 Постановка задачи проектирования

Разрабатываемая база данных будет предназначена для реализации веб-приложения, автоматизирующего работу корпуса общественных наблюдателей Российского союза молодежи Курской области.

База данных будет содержать информацию о пользователях, доступных пунктах проведения экзаменов, экзаменах, загруженных на платформу файлах, занятиях, оценках пользователей, плане проверок экзаменов, отчетах о проверке.

## 2.2. Описание предметной области

Заказчик информационной системы, использующей базу данных - корпус общественных наблюдателей РСМ Курской области (далее - КОН РСМ). КОН РСМ осуществляет проверку проведения государственных экзаменов для выпускников школ на наличие нарушений.

Корпус общественных наблюдателей набирает наблюдателей из числа желающих, после чего проводит обучение кандидатов в течение 12 занятий. Занятия проводят руководители корпуса и в ходе занятий используют систему электронного тестирования(инструмент Google forms). После 12 занятий проводится экзаменационное тестирование, по результатам которого определяется список наблюдателей, который будут допущены до работы. На наблюдателей заполняется пакет документов, по установленной форме с использованием личных данных. Корпус наблюдателей формируется список пунктов проведения экзаменов, с определенными датами и экзаменами, на каждый экзамен в ППЭ в день проведения экзамена прибывает наблюдатель(иногда не один) и в течение дня заполняет отчет о наблюдении. Отчет заполняется постепенно, в течение дня заполняются определенные пункты. Некоторые из пунктов заполняются в соответствии с документами, регламентирующими работу наблюдателя. В конце дня отчет сдается руководителю на проверку и сбор нужной информации из отчета.

Требуемыми функциями системы, использующей базу данных, являются:

1. регистрация всех членов КОН и хранение их персональных данных
2. Загрузка общей базы файлов и личных файлов пользователей
3. Создание списка ППЭ, экзаменов
4. Создание плана проведения занятий и централизованная оценка успеваемости обучающихся
5. Создание плана проверки ППЭ и прикрепление отчетов о проверке к плану
6. Создание всей необходимой документации в автоматическом режиме

Актуальность автоматизации работы КОН РСМ заключается в ускорении создания документации, безошибочности ее заполнения, в централизованном месте обучения кандидатов, упрощении работы наблюдателей на пунктах проведения экзаменов – исключение человеческого фактора, централизованное хранение информации, удобный формат обработки полученных данных.

**Описание понятий предметной области:**

1. КОН РСМ – Корпус общественных наблюдателей Российского союза молодежи

* ФОН – Федеральный общественный наблюдатель.
* Пользователь – федеральный общественный наблюдатель.
* ППЭ – пункт проведения экзамена.
* Экзамен - обязательная государственная итоговая аттестация (далее – ГИА). Формы проведения ГИА по образовательным программам основного общего образования – основной государственный экзамен (ОГЭ) и государственный выпускной экзамен (ГВЭ) , ГИА по образовательным программам среднего общего образования, итоговое сочинение (изложение).
* Проверка – присутствие ФОН в определенном ППЭ в день проведения экзамена.
* Отчет - это оформленная информация о прохождении экзамена в ППЭ.
* Документ - это файл в формате PDF, docx, xlsx, jpg и других.

Основываясь на описании предметной области выделены такие сущности:

1) Пункт проведения экзамена (ППЭ)

2) Экзамен

3) Строка плана проверки

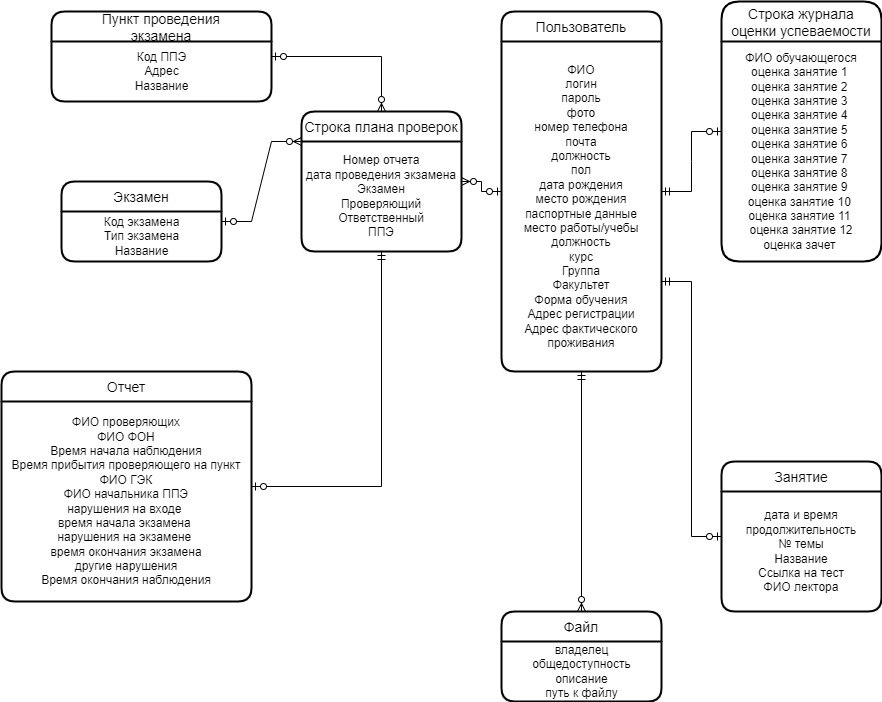
4) Отчет

5) Пользователь

6) Файл

7) Строка журнала успеваемости

8) Занятие

**Построена концептуальная схема базы данных( Рисунок 2.1.):**

**Рисунок 2.1.** Концептуальная схема базы данных

## 2.3 Анализ аналогов.

Аналогов на момент создания системы нет.

## 2.4 Перечень задач, подлежащих решению в процессе разработки.

* Разработка компонентов пользовательского интерфейса;
* Разработка структуры базы данных;
* Разработка архитектуры приложения;
* Разработка клиентской части;
* Разработка серверной части приложения;

## 2.5 Обоснование выбора инструментов и платформы для разработки.

В качестве среды разработки используется Visual Studio Code. Интегрированная среда разработки Visual Studio Code — это стартовая площадка для написания, отладки и сборки кода, а также последующей публикации приложений. Интегрированная среда разработки (IDE) представляет собой многофункциональную программу, которую можно использовать для различных аспектов разработки программного обеспечения. Помимо стандартного редактора и отладчика, которые существуют в большинстве сред IDE, Visual Studio Code включает в себя компиляторы, средства автозавершения кода, графические конструкторы и многие другие функции для упрощения процесса разработки.[1]

**Visual Studio Code[2]** — это сервис, который позиционируется как «легкий» редактор кода для кроссплатформенной разработки веб- и облачных приложений.

### Особенности ****Visual Studio Code****

* VS Code позволяет разрабатывать как консольные приложения, так и приложения с графическим интерфейсом, в том числе с поддержкой технологии Windows Forms, а также веб-сайты, веб-приложения, веб-службы как в родном, так и в управляемом кодах для всех платформ.
* В редакторе присутствуют встроенный отладчик, инструменты для работы с Git и средства рефакторинга, навигации по коду, автодополнения типовых конструкций и контекстной подсказки.
* Продукт поддерживает разработку для платформ ASP.NET и Node.js, и считается легковесным решение, которое позволяет обойтись без полной интегрированной среды разработки.
* Большим плюсом редактора является поддержка большого количества языков, таких как C++, C#, Python, PHP, JavaScript и других.

**Преимущества Visual Studio Code**

* Множество настроек (как всей программы, так и интерфейса);
* Расширяемая библиотека дополнений и готовых решений;
* Мультифункциональность (редактор поддерживает почти все языки, используемые для создания приложений);
* Простота и гибкость.

**Для разработки проекта будут использоваться следующие инструменты:**

* язык гипертекстовой разметки HTML;

HTML[3] (от англ. HyperText Markup Language — «язык гипертекстовой разметки») — стандартизированный язык разметки документов для просмотра веб-страниц в браузере. Веб-браузеры получают HTML документ от сервера по протоколам HTTP/HTTPS или открывают с локального диска, далее интерпретируют код в интерфейс, который будет отображаться на экране монитора.

Элементы HTML являются строительными блоками HTML страниц. С помощью HTML разные конструкции, изображения и другие объекты такие как интерактивная веб-формы могут быть встроены в отображаемую страницу. HTML предоставляет средства для создания заголовков, абзацев, списков, ссылок, цитат и других элементов. Элементы HTML выделяются тегами, записанными с использованием угловых скобок. Такие теги, как <img /> и <input />, напрямую вводят контент на страницу. Другие теги, такие как <p>, окружают и оформляют текст внутри себя и могут включать другие теги в качестве под элементов. Браузеры не отображают HTML-теги, но используют их для интерпретации содержимого страницы.

В HTML можно встроить программный код на языке программирования JavaScript, для управления поведением и содержанием веб-страниц. Также включение CSS в HTML описывает внешний вид и макет страницы.

* Язык программирования JavaScript[4]

Мультипарадигменный язык программирования. Поддерживает объектно-ориентированный, императивный и функциональный стили. Является реализацией спецификации ECMAScript. JavaScript обычно используется как встраиваемый язык для программного доступа к объектам приложений. Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности веб-страницам. Основные архитектурные черты: динамическая типизация, слабая типизация, автоматическое управление памятью, прототипное программирование, функции как объекты первого класса. На JavaScript оказали влияние многие языки, при разработке была цель сделать язык похожим на Java. Языком JavaScript не владеет какая-либо компания или организация, что отличает его от ряда языков программирования, используемых в веб-разработке

* каскадные таблицы стилей CSS;

CSS используется создателями веб-страниц для задания цветов, шрифтов, стилей, расположения отдельных блоков и других аспектов представления внешнего вида этих веб-страниц. Основной целью разработки CSS являлось отделение описания логической структуры веб-страницы (которое производится с помощью HTML или других языков разметки) от описания внешнего вида этой веб-страницы (которое теперь производится с помощью формального языка CSS). Такое разделение может увеличить доступность документа, предоставить большую гибкость и возможность управления его представлением, а также уменьшить сложность и повторяемость в структурном содержимом.

Кроме того, CSS позволяет представлять один и тот же документ в различных стилях или методах вывода, таких как экранное представление, печатное представление, чтение голосом (специальным голосовым браузером или программой чтения с экрана) или при выводе устройствами, использующими шрифт Брайля.

* язык программирования PHP;

PHP[5][6] (англ. PHP: Hypertext Preprocessor — «PHP: препроцессор гипертекста»; первоначально PHP/FI (Personal Home Page / Form Interpreter), а позже названный Personal Home Page Tools — «Инструменты для создания персональных веб-страниц») — скриптовый язык общего назначения, интенсивно применяемый для разработки веб-приложений. В настоящее время поддерживается подавляющим большинством хостинг-провайдеров и является одним из лидеров среди языков, применяющихся для создания динамических веб-сайтов.

Язык и его интерпретатор (Zend Engine) разрабатываются группой энтузиастов в рамках проекта с открытым кодом. Проект распространяется под собственной лицензией, несовместимой с GNU GPL.

В области веб-программирования, в частности серверной части, PHP — один из популярных сценарных языков (наряду с JSP, Perl и языками, используемыми в ASP.NET).

Популярность в области построения веб-сайтов определяется наличием большого набора встроенных средств и дополнительных модулей для разработки веб-приложений. Основные из них:

* автоматическое извлечение POST- и GET-параметров, а также переменных окружения веб-сервера в предопределённые массивы;
* взаимодействие с большим количеством различных систем управления базами данных через дополнительные модули (MySQL, MySQLi, SQLite, PostgreSQL, Oracle Database (OCI8), Microsoft SQL Server, Sybase, ODBC, mSQL, IBM DB2, Cloudscape и Apache Derby, Informix, Ovrimos SQL, Lotus Notes, DB++, DBM, dBase, DBX, FrontBase, FilePro, Ingres II, SESAM, Firebird и InterBase, Paradox File Access, MaxDB, интерфейс PDO, Redis);
* автоматизированная отправка HTTP-заголовков;
* работа с HTTP-авторизацией;
* работа с cookies и сессиями;
* работа с локальными и удалёнными файлами, сокетами;
* обработка файлов, загружаемых на сервер;
* работа с XForms.

По состоянию на 2010-е годы используется сотнями тысяч разработчиков; согласно рейтингу корпорации TIOBE, базирующемуся на данных поисковых систем, в мае 2016 года PHP находился на 6 месте среди языков программирования.

Входит в LAMP — распространённый набор программного обеспечения для создания и хостинга веб-сайтов (Linux, Apache, MySQL, PHP).

Среди сайтов, использующих PHP — Facebook, Wikipedia, Yahoo!, Baidu.

* база данных Oracle Database 10g Express Edition[7];

Oracle Database 10g Express Edition (Oracle Database XE) - это небольшая база данных начального уровня, основанная на базе кода Oracle Database 10g Release 2, которую можно: бесплатно разрабатывать, развертывать и распространять, быстро скачать. Oracle Database XE - отличная стартовая база данных для:

* Разработчиков, работающих над приложениями PHP, Java, .NET, XML и Open Source;
* Администраторов баз данных, которым нужна бесплатная начальная база данных для обучения и развертывания;
* Независимых поставщиков программного обеспечения (ISV) и поставщиков оборудования, которые хотят, чтобы начальная база данных распространялась бесплатно.
* Образовательным учреждениям и студентам, которым нужна бесплатная база данных для их учебной программы

С Oracle Database XE можно разрабатывать и развертывать приложения с мощной, проверенной и ведущей в отрасли инфраструктурой, а затем при необходимости обновлять их без дорогостоящих и сложных миграций.

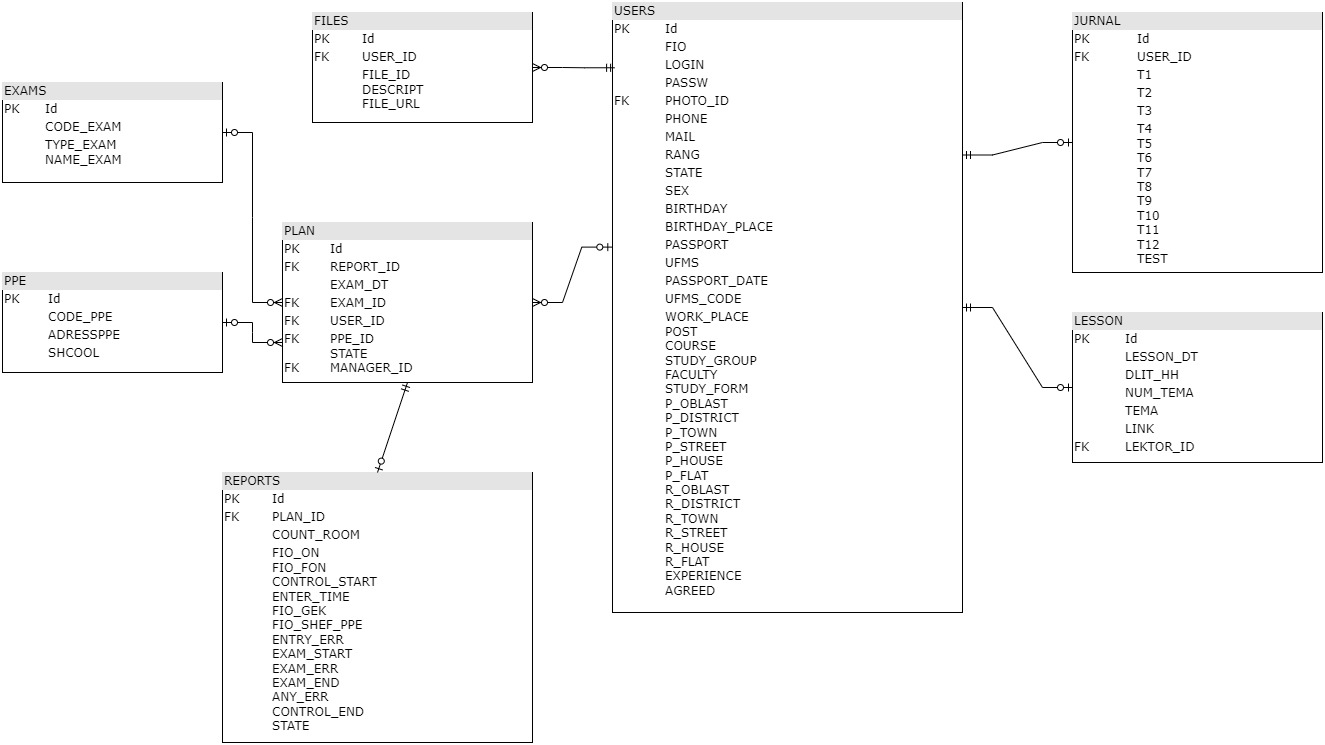
Oracle Database XE можно установить на хост-машину любого размера с любым количеством процессоров (одна база данных на машину), но XE будет хранить до 4 ГБ пользовательских данных, использовать до 1 ГБ памяти и использовать один ЦП на хост-машине.

Безусловным плюсом является наличие панели управления базой данных в браузере, не требующая установки каких-либо приложений.

### 3. Проектно- конструкторская часть

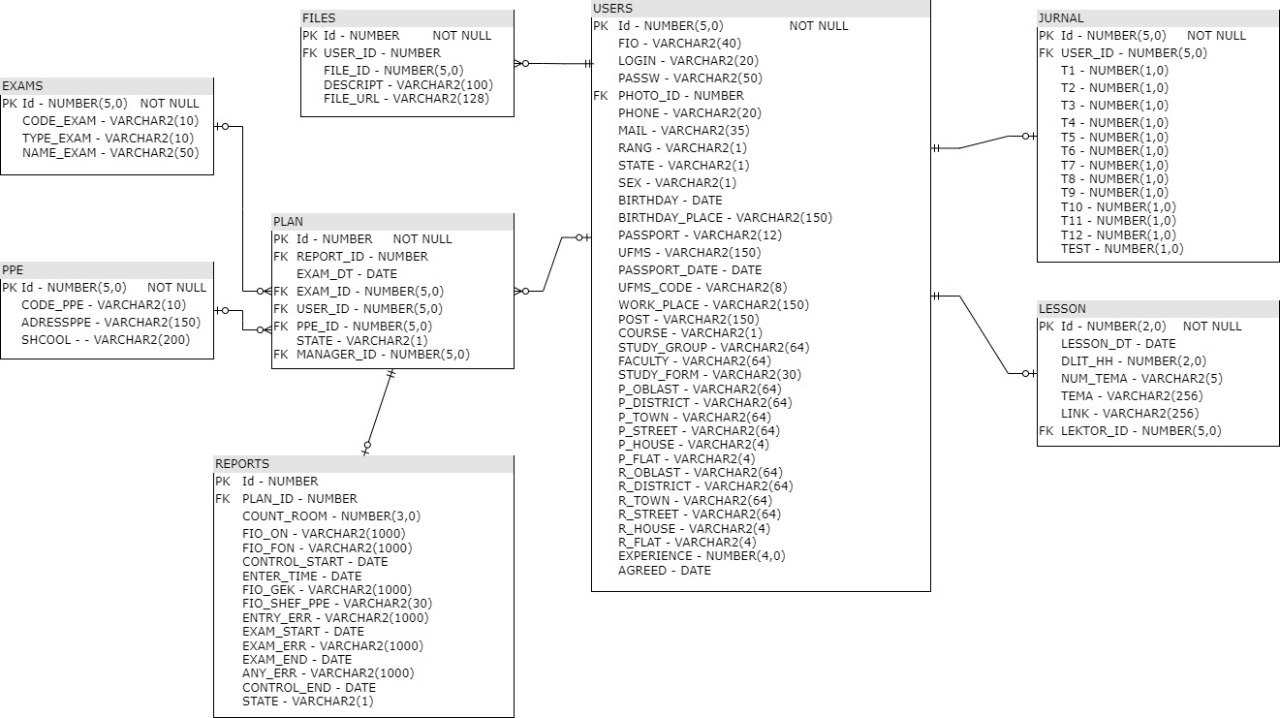
### 3.1 Логическая схема базы данных

Исходя из концептуальной модели данных спроектирована логическая модель. В логической модели данных новых сущностей не появилось, так как в концептуальной модели отсутствовали сущности со связью многие ко многим. В связи с этим логическая модель имеет вид (Рисунок 3.1.):

Рисунок 3.1. Логическая модель.

### 3.2 Физическая схема базы данных

Учитывая особенности базы данных Oracle и логическую схему проектируем физическую модель базы данных (Рисунок 3.2.):

Рисунок 3.2. Физическая модель.

### 3.3 Описание сущностей базы данных

**«EXAMS» - сущность экзаменов.** Используется для описания предмета, сдаваемого на итоговой государственной аттестации.

Имеет поля:

Id - NUMBER(5,0) NOT NULL - искусственный первичный ключ

CODE\_EXAM - VARCHAR2(10) - Код экзамена

TYPE\_EXAM - VARCHAR2(10) - Тип экзамен(ЕГЭ, ОГЭ и тд)

NAME\_EXAM - VARCHAR2(50) - Название экзамена

**«PPE» - сущность пунктов проведения экзаменов.** Используется для описания пункта проведения экзамена (ППЭ), то есть пункта организованного на базе школы.

Имеет поля:

Id - NUMBER(5,0) NOT NULL - искусственный первичный ключ

CODE\_PPE - VARCHAR2(10) - код пункта проведения экзамена

ADRESSPPE - VARCHAR2(150) - адрес пункта проведения экзамена

SHCOOL - VARCHAR2(200) - название общеобразовательного учреждения

**«REPORTS» - сущность отчетов.** Используется для описания отчета о проверке нарушений проведения экзамена.

Имеет поля:

Id – NUMBER – искусственный первичный ключ

PLAN\_ID – NUMBER – внешний ключ в поле Id таблицы PLAN

COUNT\_ROOM - NUMBER(3,0) – количество аудиторий

FIO\_ON - VARCHAR2(1000) – ФИО общественных наблюдателей

FIO\_FON - VARCHAR2(1000) – ФИО федеральных общественных наблюдателей

CONTROL\_START – DATE - время начала контроля

ENTER\_TIME – DATE – время прихода наблюдателя в ППЭ

FIO\_GEK - VARCHAR2(1000) – ФИО ГЭК

FIO\_SHEF\_PPE - VARCHAR2(30) – ФИО начальника ППЭ

ENTRY\_ERR - VARCHAR2(1000) – Нарушения на входе в ППЭ

EXAM\_START – DATE – Время начала экзамена

EXAM\_ERR - VARCHAR2(1000) – Нарушения во время проведения экзамена

EXAM\_END – DATE – Время окончания экзамена

ANY\_ERR - VARCHAR2(1000) – Иные нарушения

CONTROL\_END – DATE – время окончания наблюдения

STATE - VARCHAR2(1) – состояние отчета

**«PLAN» - сущность записей плана проверок ППЭ.** Используется для описания задачи проверки: какой наблюдатель, на какой экзамен, в какой пункт проведения, в какой день отправляется проверять. По результатам проверки создается отчет, прикрепляющийся к определенной записи плана.

Имеет поля:

Id – NUMBER – искусственный первичный ключ

REPORT\_ID – NUMBER – внешний ключ в поле Id таблицы REPORTS

EXAM\_DT – DATE – дата проведения экзамена

EXAM\_ID - NUMBER(5,0) – внешний ключ в поле Id таблицы EXAMS

USER\_ID - NUMBER(5,0) - внешний ключ в поле Id таблицы USERS

PPE\_ID - NUMBER(5,0) - внешний ключ в поле Id таблицы PPE

STATE - VARCHAR2(1) - состояние

MANAGER\_ID - NUMBER(5,0) - внешний ключ в поле Id таблицы USERS

**«FILES» - сущность файлов.** Используется для описания файлов, загруженных в систему, физически загруженных в файловую систему сервера.

Имеет поля:

Id – NUMBER - искусственный первичный ключ

USER\_ID – NUMBER – внешний ключ в поле Id таблицы USERS

FILE\_ID - NUMBER(5,0) – идентификатор файла

DESCRIPT - VARCHAR2(100) – описание файла

FILE\_URL - VARCHAR2(128) – адрес файла в файловой системе

**«USERS» - сущность пользоватлей.** Используется для описания пользователей и администраторов системы (Общественных наблюдателей и руководства корпуса общественных наблюдателей).

Имеет поля:

Id - NUMBER(5,0) - искусственный первичный ключ

FIO - VARCHAR2(40) – ФИО пользователя

LOGIN - VARCHAR2(20) – Логин пользователя

PASSW - VARCHAR2(50) – Пароль пользователя

PHOTO\_ID – NUMBER – внешний ключ в поле Id таблицы FILES

PHONE - VARCHAR2(20) – номер телефона пользователя

MAIL - VARCHAR2(35) – email пользователя

RANG - VARCHAR2(1) – ранг пользователя внутри системы

STATE - VARCHAR2(1) – состояние пользователя

SEX - VARCHAR2(1) - пол

BIRTHDAY – DATE – дата рождения пользователя

BIRTHDAY\_PLACE - VARCHAR2(150) – место рождения

PASSPORT - VARCHAR2(12) – серия и номер паспорта

UFMS - VARCHAR2(150) – кем выдан паспорт

PASSPORT\_DATE – DATE – дата выдачи паспорта

UFMS\_CODE - VARCHAR2(8) – код подразделения, выдавшего паспорт

WORK\_PLACE - VARCHAR2(150) – место работы/ учебы

POST - VARCHAR2(150) – занимаемая должность

COURSE - VARCHAR2(1) – курс обучения

STUDY\_GROUP - VARCHAR2(64) – группа обучения

FACULTY - VARCHAR2(64) - факультет

STUDY\_FORM - VARCHAR2(30) – форма обучения

P\_OBLAST - VARCHAR2(64) – облась проживания по прописке

P\_DISTRICT - VARCHAR2(64) – район проживания по прописке

P\_TOWN - VARCHAR2(64) – город проживания по прописке

P\_STREET - VARCHAR2(64) – улица проживания по прописке

P\_HOUSE - VARCHAR2(4) – дом проживания по прописке

P\_FLAT - VARCHAR2(4) – квартира проживания по прописке

R\_OBLAST - VARCHAR2(64) – область фактического проживания

R\_DISTRICT - VARCHAR2(64) – район фактического проживания

R\_TOWN - VARCHAR2(64) – город фактического проживания

R\_STREET - VARCHAR2(64) – улица фактического проживания

R\_HOUSE - VARCHAR2(4) – дом фактического проживания

R\_FLAT - VARCHAR2(4) – квартира фактического проживания

EXPERIENCE - NUMBER(4,0) – год вступления в КОН

AGREED – DATE – согласие на обработку персональных данных

**«LESSON» - сущность занятий.**  Описывает занятие, проводимое с общественными наблюдателями в течение их обучения.

Имеет поля:

Id - NUMBER(2,0) - искусственный первичный ключ

LESSON\_DT – DATE – дата и время проведения занятия

DLIT\_HH - NUMBER(2,0) – длительность занятия

NUM\_TEMA - VARCHAR2(5) – номер темы

TEMA - VARCHAR2(256) – название темы

LINK - VARCHAR2(256) – ссылка на сторонний ресурс

LEKTOR\_ID - NUMBER(5,0) – внешний ключ в поле Id таблицы USERS

**«JURNAL» - сущность журнала.** Используется для хранения оценок каждого пользователя по всем занятиям.

Имеет поля:

Id - NUMBER(5,0) – искусственный первичный ключ

USER\_ID - NUMBER(5,0) ) – внешний ключ в поле Id таблицы USERS

T1 - NUMBER(1,0) – оценка за занятие 1

T2 - NUMBER(1,0) – оценка за занятие 2

T3 - NUMBER(1,0) – оценка за занятие 3

T4 - NUMBER(1,0) – оценка за занятие 4

T5 - NUMBER(1,0) – оценка за занятие 5

T6 - NUMBER(1,0) – оценка за занятие 6

T7 - NUMBER(1,0) – оценка за занятие 7

T8 - NUMBER(1,0) – оценка за занятие 8

T9 - NUMBER(1,0) – оценка за занятие 9

T10 - NUMBER(1,0) – оценка за занятие 10

T11 - NUMBER(1,0) – оценка за занятие 11

T12 - NUMBER(1,0) – оценка за занятие 12

TEST - NUMBER(1,0) – оценка за экзамен

### 3.4 Sequence

Из-за отсутствия в базе данный автоинкрементируемых полей, для генерации идентификаторов к каждой таблице создана последовательнось - структура для генерации уникальных целочисленных значений.

Для таблицы «EXAMS»

***CREATE SEQUENCE "SEQ\_EXAMS" MINVALUE 1 MAXVALUE 999999999999999999999999999 INCREMENT BY 1 START WITH 103 CACHE 20 NOORDER NOCYCLE***

Поледовательность от 1 до 999999999999999999999999999 с шагом в 1, начинается со значения 103, заранее создается 20 значений, для ускорения работы, параметр NOORDER указывает, что генерируемые значения не обязательно будут идти по порядку, последовательность не циклична, что приведет к ошибке при попытке запросить новый член последовательности при достижении максимального значения.

### 3.5 Хранимые процедуры и функции

Для повышения производительности и безопасности в проекте используются хранимые процедуры и функции. Производительность увеличивается за счет того, что хранимые процедуры и функции компилируются заранее и план выполнения создаетя один раз, а затем повторно использует этот план выполнения. Это приводит к повышению производительности в случае многократного вызова хранимой процедуры, так как обработчику запросов не нужно создавать новый план и поэтому обработка процедуры или функции занимает меньше времени.

Для сохранения файлов используется процедура:

***create or replace PROCEDURE save\_user\_file(p\_user\_id in number, p\_descript in VARCHAR2, p\_FileName in VARCHAR2, f\_id out number) as ST\_SQL varchar2(1000) := 'Insert into files values (SEQ\_Files.nextval,null,:1,:2,:3) returning ID into :4'; begin execute immediate ST\_SQL using p\_user\_id, p\_descript, p\_FileName, out f\_id; end save\_user\_file;***

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

В поцедуре используются 3 входных параметра и один выходной, в блоке begin выполняется динамический SQL запрос, параметры в который подставляются последовательно после ключевого слова **using.** В выходной параметр записывается идентификатор файла последством **returning**.

Для аутентификации используется функция с возвращаемым значением типа boolean.

***create or replace function password\_auth (p\_username in VARCHAR2, p\_password in VARCHAR2) return BOOLEAN is***

***l\_password varchar2(50); l\_stored\_password varchar2(50);***

***l\_count number;***

***begin select count(\*) into l\_count from users where login = p\_username and State = '1'; if l\_count > 0 then select passw into l\_stored\_password from users where login = p\_username;***

***l\_password := password\_hash(p\_username, p\_password);***

***if l\_password = l\_stored\_password then return true; else return false;***

***end if;***

***else return false;***

***end if;***

***end;***

Получаем на вход логин и пароль пользователя, проверяем на наличие в базе данных пользователей с таким логином и активным состоянием, если такие есть, то вычисляем hash из полученного пароля и сравниваем его с паролем, хранимым в базе данных. Если пароли совпадают – возвращаем true, если нет – false. False вернется и в том случае, если пользователей с таким логином не найдено.

Вызываемая **password\_hash –** тоже хранимая функция.

### 3.6 Триггеры

В базе данных также используются триггеры, некоторые для поддержания целостности базы данных, некоторые из них для реализации логики. Одним из триггеров, используемых для реализаци логики является триггер «TR\_EXAMS». Он предназначен для получения первичного ключа и сохранения его в запись перед выполнением DML оператора:

***CREATE OR REPLACE TRIGGER "TR\_EXAMS"***

***before insert on EXAMS***

***for each row***

***begin***

***select SEQ\_EXAMS.nextval into :new.ID from dual;***

***end TR\_EXAMS;***

***/***

***ALTER TRIGGER "TR\_EXAMS" ENABLE***

***/***

Для простоты аутентификации используется триггер «TR\_USERS\_2»

***CREATE OR REPLACE TRIGGER "TR\_USERS\_2"***

***BEFORE***

***insert or update on USERS***

***for each row***

***begin***

***:NEW.LOGIN := upper(:NEW.LOGIN);***

***end TR\_Users\_2;***

***/***

***ALTER TRIGGER "TR\_USERS\_2" ENABLE***

***/***

Он преобразует логин в верхний регистр. При аутентификации введенный логин тоже преобразуется в верхний регистр.

### 3.7 Нормализация базы данных

Табицы базы данных соответствуют второй нормальной форме, так как для оответствия ей таблицы должны находиться в первой нормальной форме, которая заключается в отсутствии дублирующих строк, отсутствии массивов и списков в любом виде. Дополнительно должны выполняться требования:

* Таблицы должны иметь первичный ключ;
* Все неключевые столбцы должны зависеть от полного ключа.

В таблицах используются суррогатные ключи, находящиеся в отдельном столбце, поэтому зависимости от части ключа в таблицах нет.

Третья нормальная форма подразумевает отсутствие транзитивной зависимости (неключевые столбцы зависят от значений других неключевых столбцов), такая зависимость присутствует в таблице пользователей («USERS»), в ней столбец «UFMS\_CODE» зависит от другого неключевого столбца «UFMS», так же потенциально есть зависимость столбца «STUDY\_GROUP» от столбца «FACULTY». Привести таблицы к третьей нормальной форме можно декомпозировав уже существующие таблицы, но это не рационально с точки зрения использования базы данных. Из-за частых запросов на выборку информации о пользователях нужно будет соединять три таблицы вместе, что замедлит выполнение запроса.

### 3.8 Обеспечение целостности базы данных

Целостность базы данных достигается посредством использования триггеров, работающих перед операциями удаления или изменения. Например, триггер для удаления файла из системы: удаление файлов с сервиса разрешено, но перед операцией удаления производится очистка всех полей, где находится идентификатор удаляемого файла.

Удаление строк таблиц запрещено, во избежние нарушения целостности. Удаленные строки только помечаются удаленными, что не нарушает связи, позволяет в старых записях отображать удаленные строки, а при создании новых записей не отображает удаленные строки.

### ПРОЕКТНО-ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ

### 4.1 Процедуры автоматизации для обеспечения отказоустойчивости.

Данное приложение в текущем варианте развертывания работает не на серверном оборудовании. На персональный компьютер установлен веб-сервер, на котором работает приложение. Из-за перебоев с электропитанием возможны перебои в работе сервера. Для избежания долгих отключений к персональному компьютеру подключается микроконтроллер, отслеживающий работу компьютера, который в случае обнаружения факта выключения компьютера запускает его снова. Для работы сервера требуется запуск СУБД и веб-сервера denwer. Для этого их службы прописываются в автозапуск. А для избежания потери информации из базы данных, присутствует скрипт, выполняющий резервное кпирование файлов базы данных на внешний носитель.

### 4.2 Порядок развёртывания системы.

Данное веб-приложение было развернуто на домашнем сервере с операционной системой Windows 7. Для работы с системой из сети интернет нужно сделать:

* Подключить «белый» IP-адрес;
* Установить веб-сервер denwer;
* Установить Oracle Database 10g Express Edition;
* Загрузить необходимые dll, скопировать их в каталог denwer’а, добавить расширение в файл php5.ini;
* Скопировать все файлы веб-приложения в папку www denwer’а;
* Открыть в denwer’е доступ из сети интернет;
* Утановить статический ip-адрес компьютера, на котором развернут сервер в сети маршрутизотора;
* На маршрутизаторе настроить проброс порта.

При развертывании данного веб-приложения обязательно требуется н сервере: Oracle Database 10g Express Edition, поддержка PHP5. Тогда потребуется только выполнить перенос данных в новую базу данных и перенести все файлы веб-приложения на новый сервер.

### 4.3 Разработка руководства пользователя и руководства администратора.

Руководство пользователя не создается в связи с простотой и интуитивностью интерфейса.

**Руководство администратора:**

1. Администраторы входят в систему с общей страницы авторизации(Рисунок 4.3.1). Администраторы вводят свой логин и пароль. Права администратора можно выдать при создании пользователя, либо изменить свойства пользователя в панели работы с пользователями.

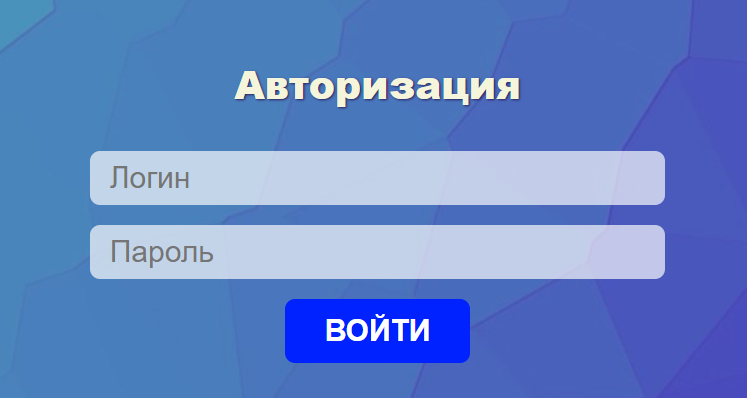


Рисунок 4.3.1 – Форма авторизации

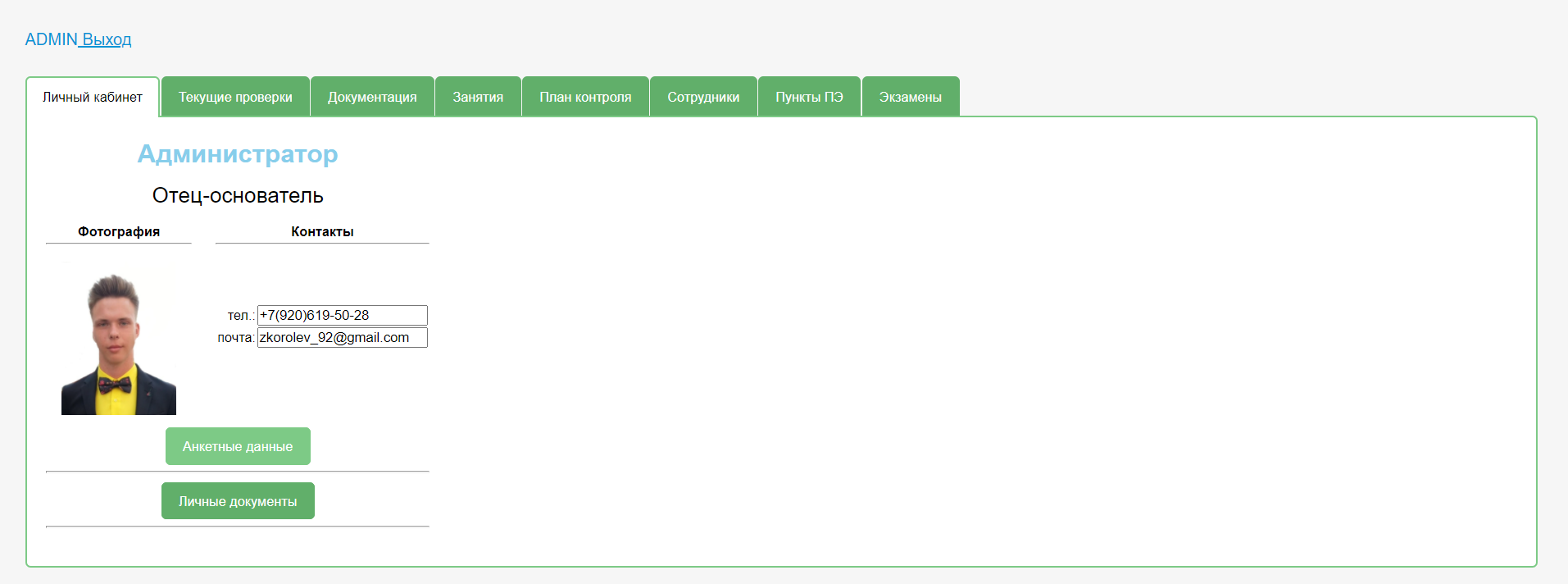
1. После авторизации происходит перенаправление на страницу администраторов. (Рисунок 4.3.2). В верхней части страницы отобразятся все доступные вкладки для администраторов. При нажатии кнопки «Анкетные данные» открывается анкета пользователя, после изменения данных требуется нажать кнопку «Сохранить». При нажатии кнопки «Личные документы» откроется список документов текущего пользователя, их можно добавить, удалить, либо выбрать главное фото.

Рисунок 4.3.2 – Интерфейс администраторов.

1. Во вкладке «Текущие проверки» отображается список плановых проверок федеральных общественных наблюдателей (пользователей) за которых он ответственен (Рисунок 4.3.3). Администротору будут достпуны все отчеты пользователей за экзамены, которые уже прошли, либо идут в текущее время. При нажатии кнопки «GoogleForm» происходит перенаправление на форму, которую администратор обязан заполнять каждый день, когда у его подчиненных есть проверяемый экзамен.

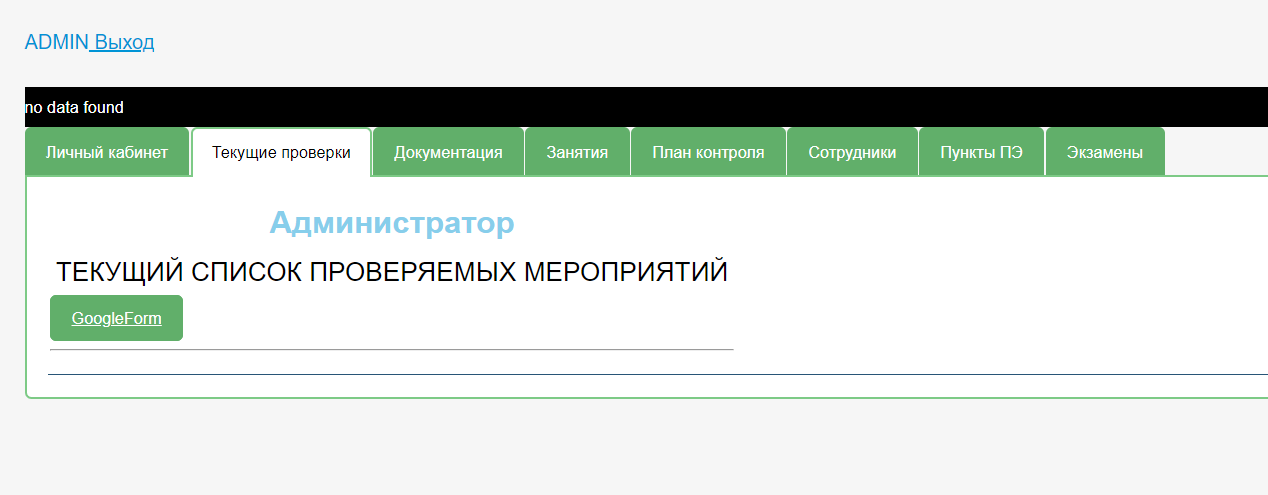


Рисунок 4.3.3 – Страница текущие проверки

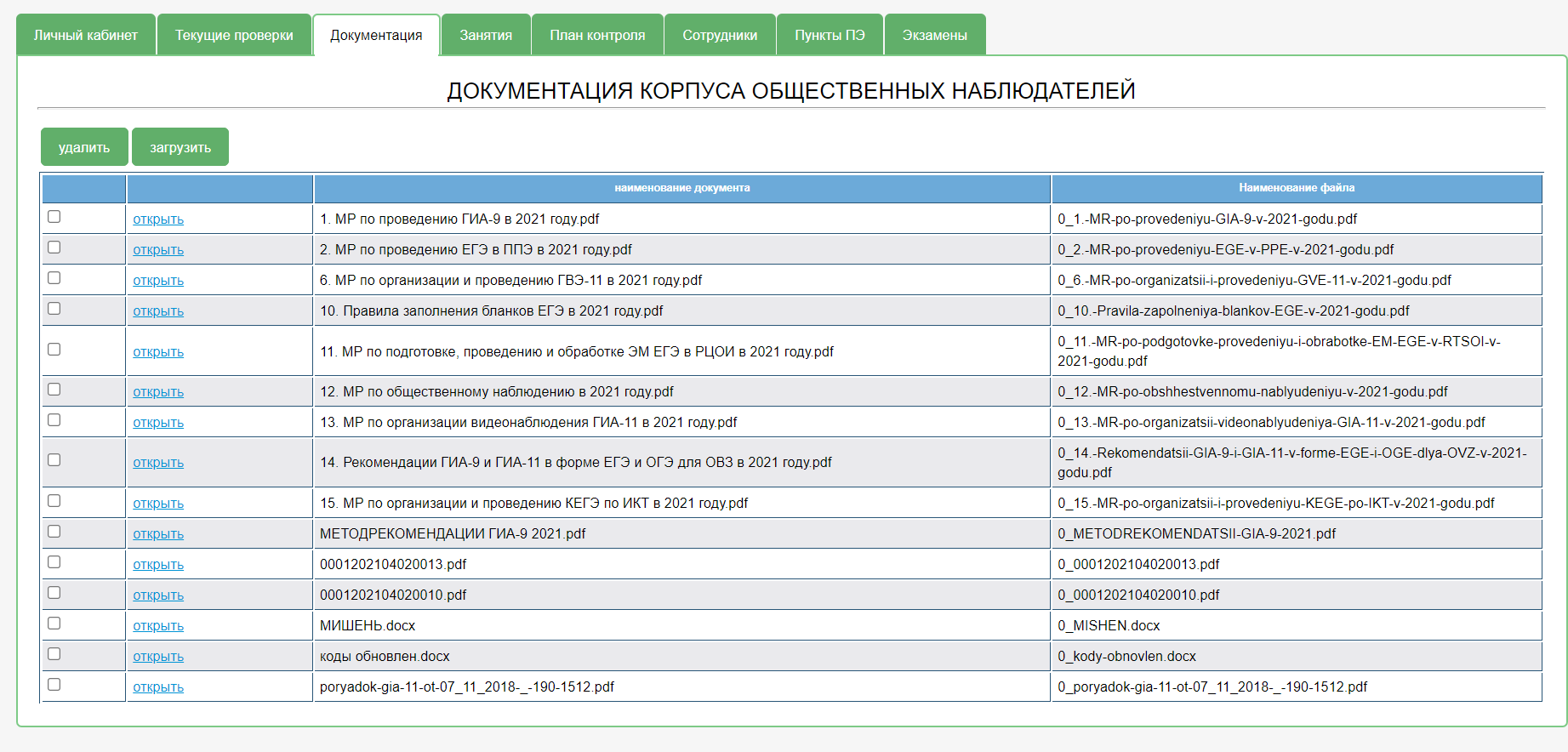
1. Во владке документация (Рисунок 4.3.4) – список всех общедоступных документов. Эти документы загружать и удалять могут только администраторы. Загрузка файла происходит при нажатии на кнопку «Загрузить», удалить файл/ файлы можно выбрав нужные, поставив галочки в чекбоксы в превом столбце и нажав кнопку «Удалить». При нажатии кнопки «Открыть» в строке с интересующим файлом происходит либо открытие файла в браузере, если такой формат поддерживается браузером, либо загрузка файла на усройство пользователя/ администратора.

Рисунок 4.3.4 – Страница со списком общедоступных документов

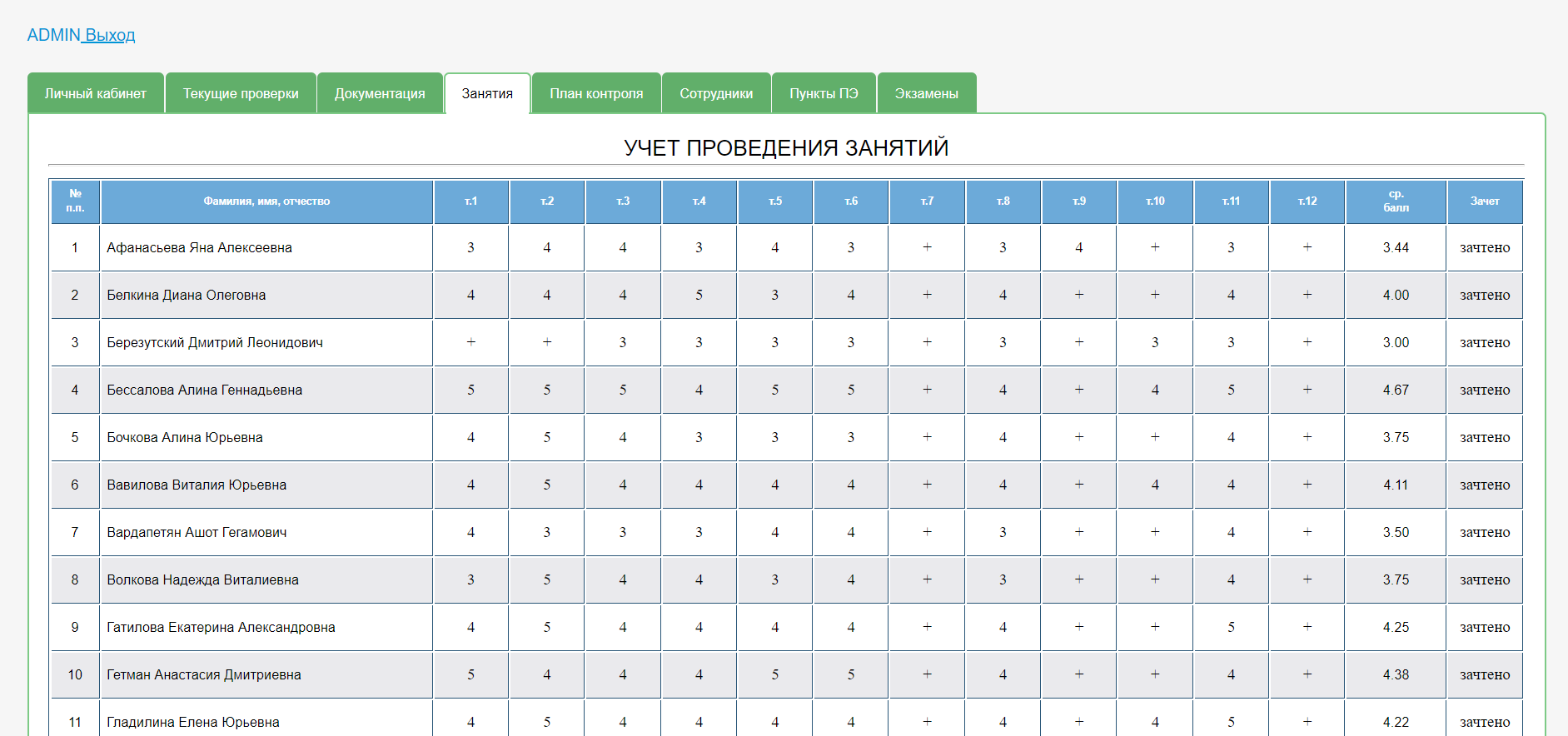
1. Страница «Занятия» (Рисунок 4.3.5). На этой странице отображается журнал всех пользователей, зарегистрированных в системе и проходящих обязательное обучение в количестве 12 занятий и зачета. При нажатии кнопки «Ведомость» (Рисунок 4.3.6) происходит формирование отчетной ведомости по оценкам всех пользователей, проходящих обучение в формате .xlsx и загрузка этого документа на устройство администратора. Чтобы выставить оценку в журнал, нужно нажать в интересующую ячейку и в выпадающем списке выбрать нужную оценку (Рисунок 4.3.7). Под журналом распологается список самих занятий (Рисунок 4.3.8), в котором можно редактировать дату и время проводимого занятия, продолжительность занятия, тему, ссылку на сторонний материал для занятия и руководителя самого занятия. Перейти в режим редактирования можно нажав на иконку пишущей руки в первом столбце интересующего занятия. Затем будет открыт интерфейс редактирования занятия (Рисунок 4.3.9). После окончания редактирования для внесения изменений в базу данных требуется нажать кнопку «Сохранить».

Рисунок 4.3.5 – Страница занятий. Вид журнала.

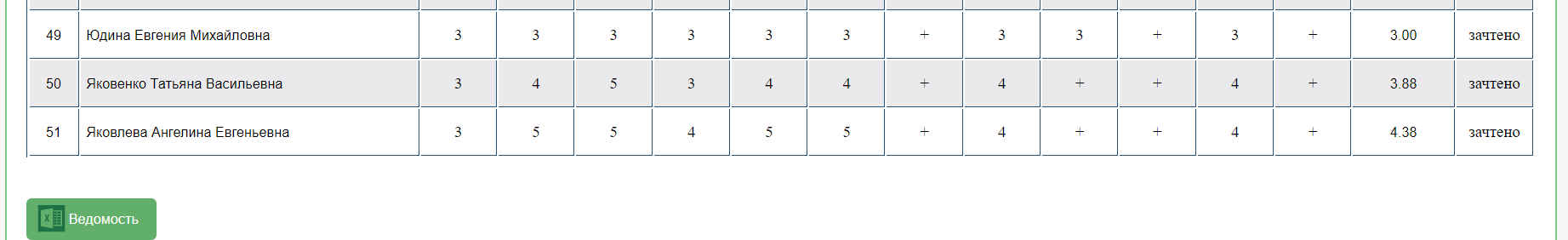


Рисунок 4.3.6 – Кнопка загрузки ведомости



Рисунок 4.3.7 – Выставление оценки



Рисунок 4.3.8 – Список занятий

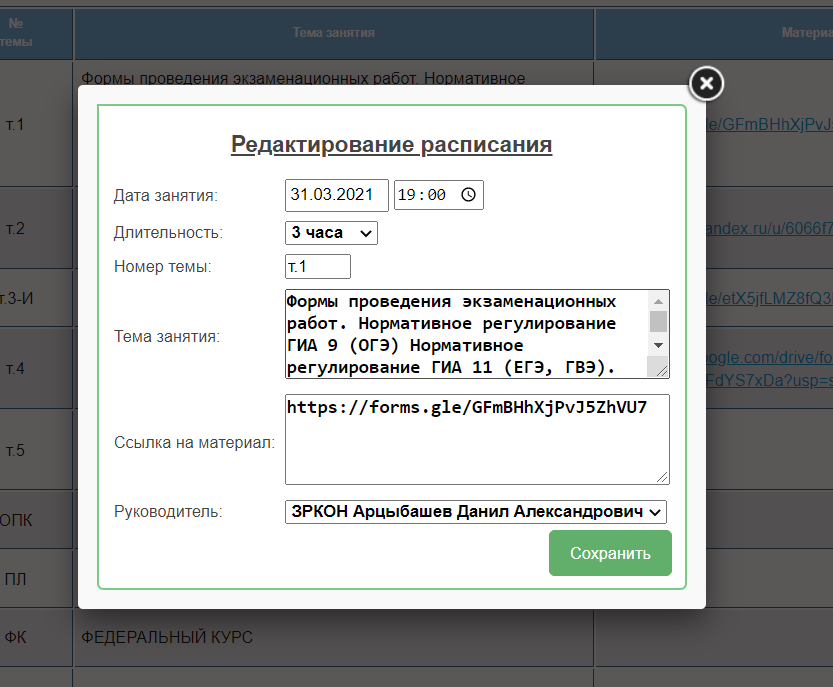


Рисунок 4.3.9 – Интерфейс изменения занятия

1. План контроля – на этой вкладке отображается план проверки пунктов проведения экзаменов федеральными общественными наблюдателями. Имеется возможность изменять каждое поле определенной записи. На странице имеется возможность установить границы отображения плана – «Выписка за период». Помимо этого существует возможность загрузки отчета – это полная ведомость всех записей плана в заданном промежутке, сформированная в формате .xlsx. При нажатии кнопки «Отчет» происходит формирование отчета и загрузка на устройство пользователя. При нажатии кнопки «Добавить» происходит добавление записи в план. Все поля редактируются нажатием на нужное поле.



Рисунок 4.3.10 – План проверки

1. Вкладка сотрудники содержит в себе список всех зарегистрированных в системе пользователей. Поле «Населенный пункт» заполнять следует при формировании пакета документов на определенного пользователя. «Поиск» осуществляет выборку пользователей подходящих по введенному условию. Поиск возможно производить по всем отображаемым полям, кроме полей «Ранг» и «Статус», то есть поиск возможен по полям: ФИО, логин, телефон, почта, №. В третьем столбце отображаются три кнопки, предназначенные:

* Верхняя кнопка – просмотр учетной записи, редактирование статуса, пароля, ранга и т.д. (Рисунок 4.3.12). Статус позволяет заблокировать пользователю вход на сайт без удаления пользователя. После изменений треуется нажать кнопку «Сохранить», при изменении пароля нужно нажать еще и на кнопку «Ок».
* Средняя кнопка – просмотр и изменение личной анкеты, при внесении изменений следует нажать кнопку «Соханить» (Рисунок 4.3.13);
* Нижняя кнопка – формирование пакета документов.

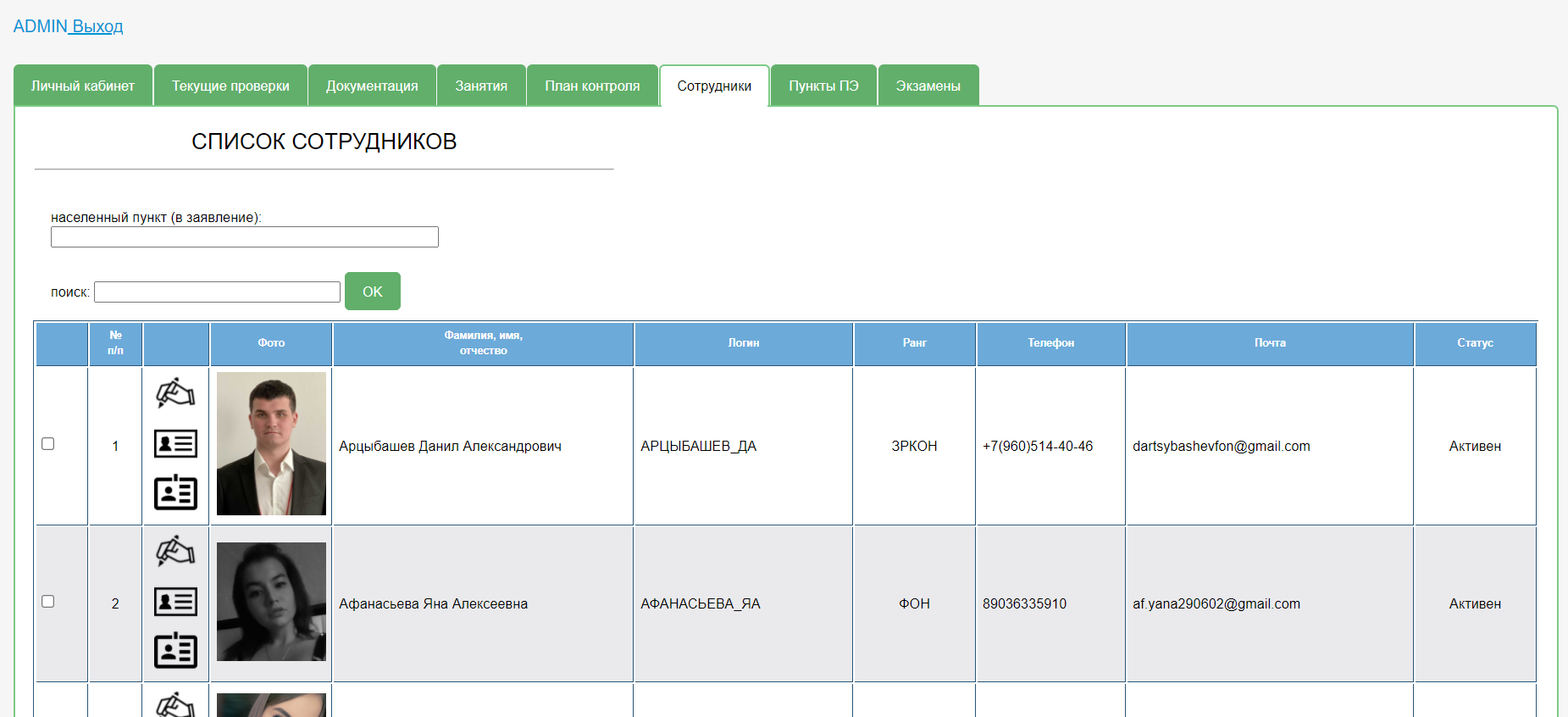


Рисунок 4.3.11 - Список сотрудников

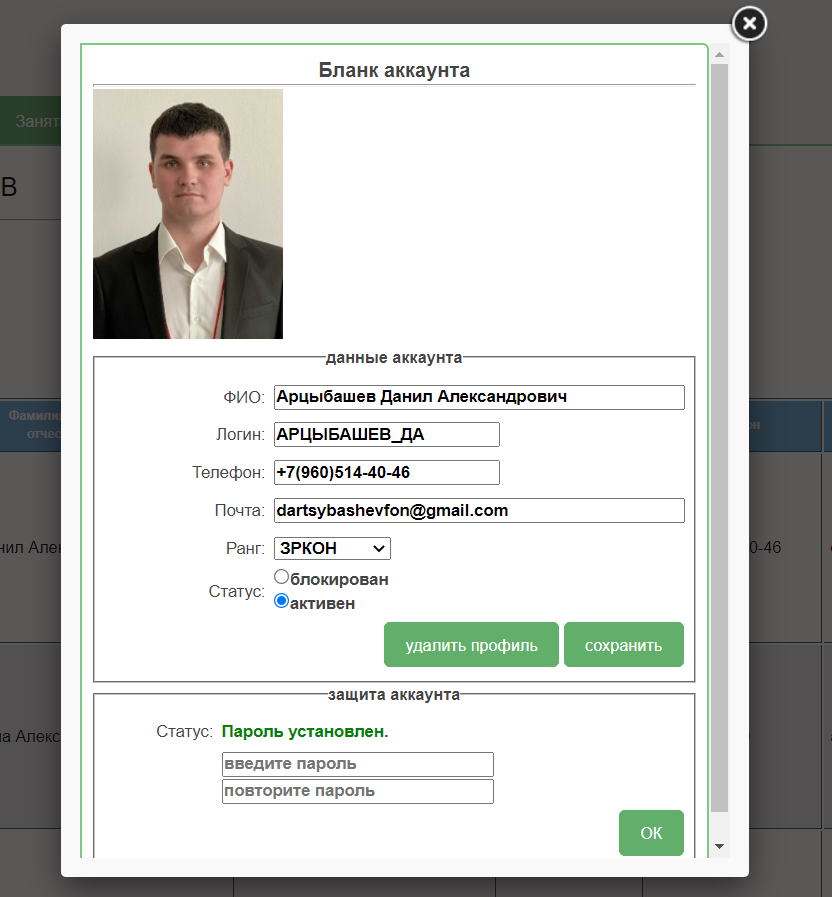


Рисунок 4.3.12 - Просмотр и редактирование учетной записи пользователя

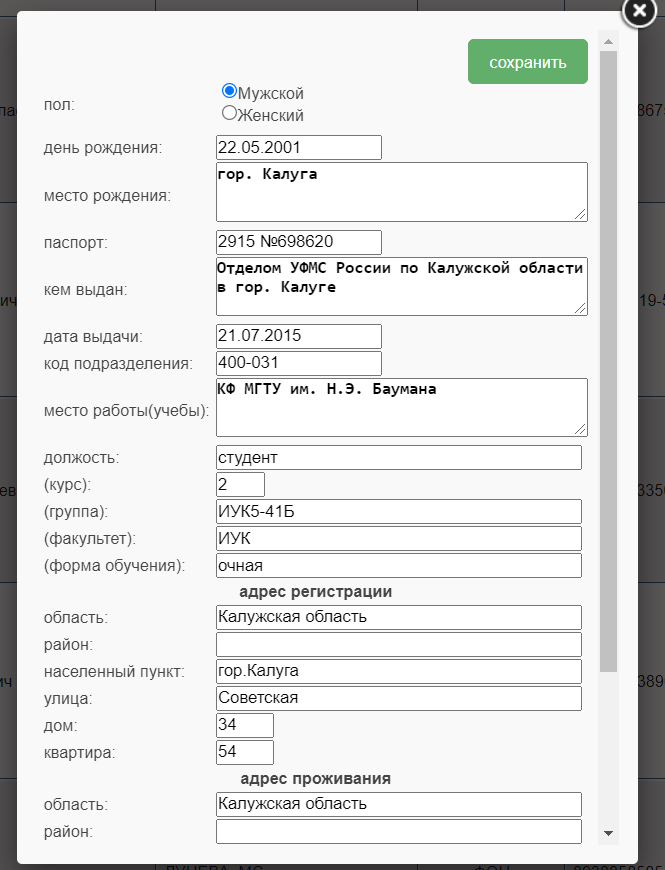


Рисунок 4.3.13 - Просмотр и редактирование анкеты пользователя

1. Вкладка ППЭ – отображает список всех пунктов проведения экзамена, в которых может проводиться проверка. Варианты редактирования описаны прямо на странице (Рисунок 4.3.14).

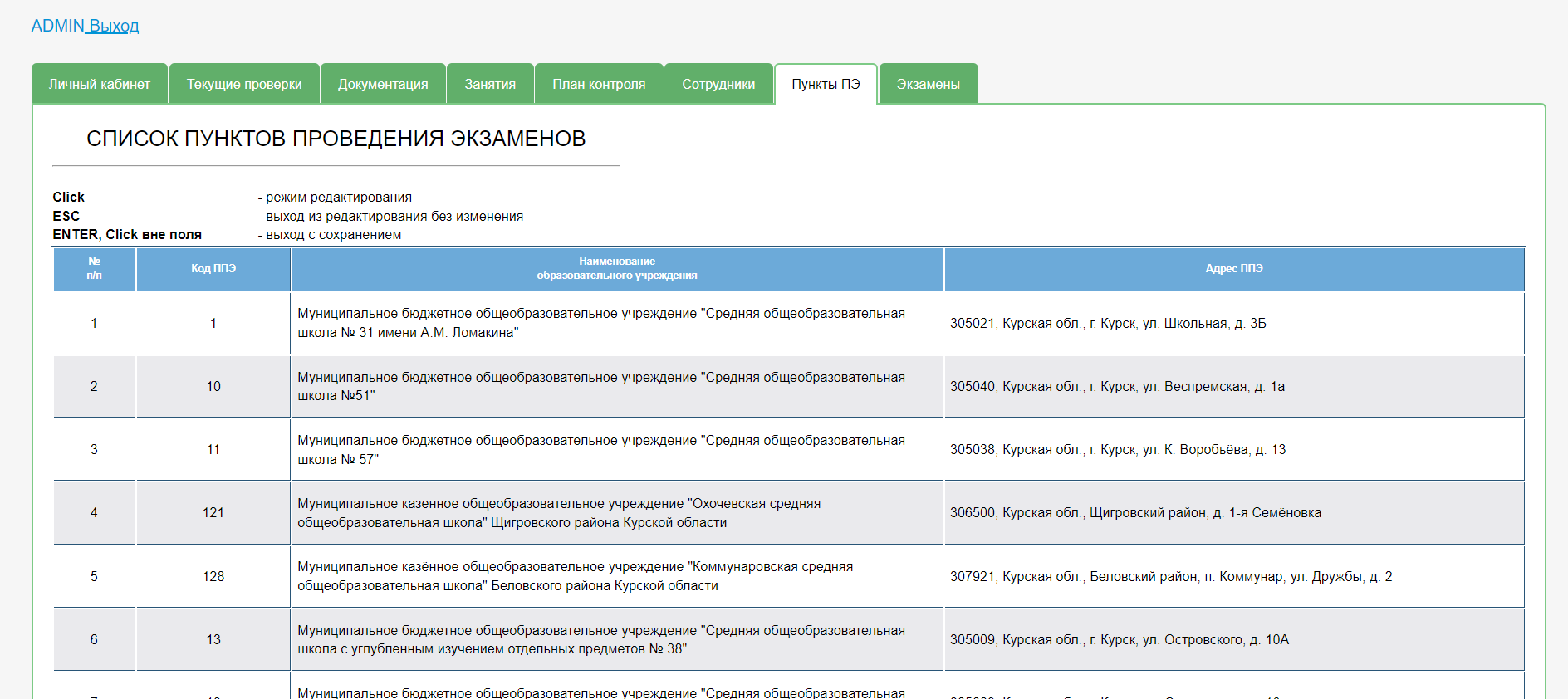


Рисунок 4.3.14 – Вкладка пунктов проведения экзамена

На этой вкладке также есть возможность создать отчет, нажав на кнопку «Отчет», и добавить новую запись, нажав на кнопку «Добавить» (Рисунок 4.3.15).

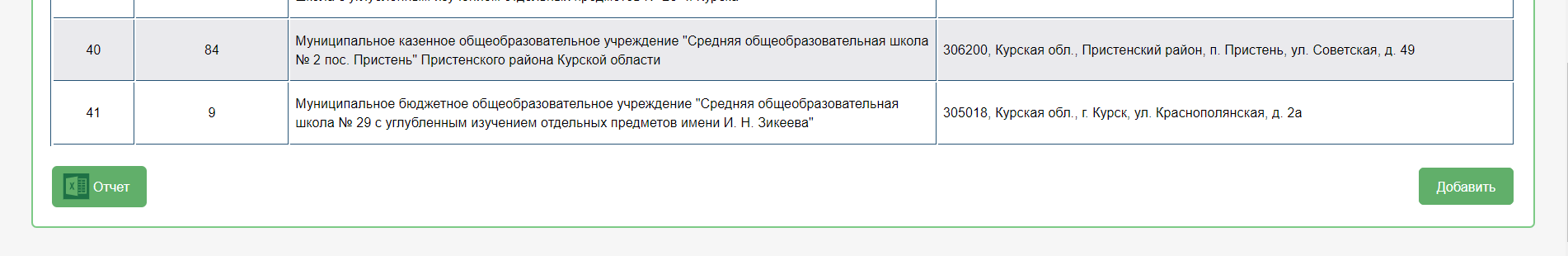


Рисунок 4.3.15 – Отчет о ППЭ и добавление новой записи

1. Вкладка Экзамены предназначена для просмотра и внесения информации об экзаменах. Функционал и управление идентичны вкладке «Пункты ПЭ»(Рисунок 4.3.16).

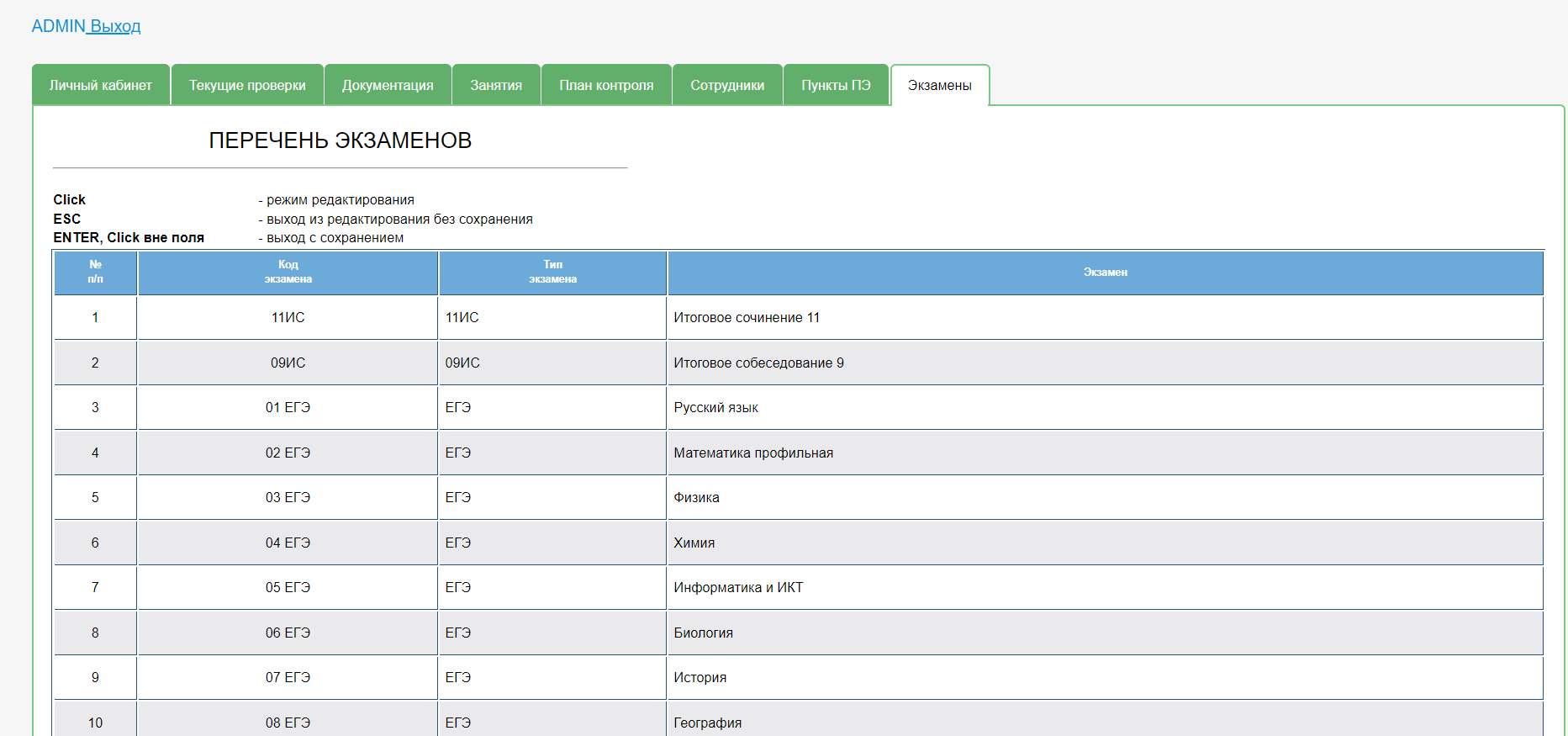


Рисунок 4.3.16 – Вкладка Экзамены

### 4.4 Тестирование и отладка рабочей программы.

Для тестирования важнейшего функционала веб-приложения были написаны и выполнены тест-кейсы.

Для тестирования функционала аутентификации тест-кейс №1(таблица 4.1 и таблица 4.2) результаты его прохождения приведены в таблице 4.3:

Таблица 4.1. Основная надпись тест-кейса номер 1

|  |  |
| --- | --- |
| **Номер тест-кейса** | 1 |
| **Заголовок** | Осуществить тестирование функционала приложения, позволяющего пользователям произвести авторизироваться |
| **Предусловие** | Перейти с главной страницы сайта на страницу «Войти» |

Таблица 4.2. Тест-кейс номер 1

|  |  |  |
| --- | --- | --- |
| **Номер шага** | **Действие** | **Ожидаемый результат** |
| 1 | Ввести логин и пароль существующего пользователя. Логин ввести в нижнем регистре. | Переход на страницу личного кабинета. |
| 2 | Ввести логин и пароль существующего пользователя. Логин ввести в верхнем регистре. | Переход на страницу личного кабинета. |

Таблица 4.2. Тест-кейс номер 1. Продолжение.

|  |  |  |
| --- | --- | --- |
| **Номер шага** | **Действие** | **Ожидаемый результат** |
| 3 | Ввести логин и пароль существующего пользователя. Логин ввести меняя регистр для разных символов. | Переход на страницу личного кабинета. |
| 4 | Ввести логин существующего пользователя. Пароль ввести неверный. | Надпись «Авторизация» заменится на «В доступе отказано, повторите» |
| 5 | Ввести логин и пароль несуществующего пользователя. | Надпись «Авторизация» заменится на «В доступе отказано, повторите» |

Таблица 4.3. Выполнение тест-кейса №1

|  |  |  |
| --- | --- | --- |
| **Шаг** | **Примечание** | **Скриншоты** |
| 1 | Выполнено, ошибок не возникло. | Отсутствует |
| 2 | Выполнено, ошибок не возникло. | Отсутствует |
| 3 | Выполнено, ошибок не возникло. | Отсутствует |
| 4 | Выполнено, ошибок не возникло. | Отсутствует |
| 5 | Выполнено, ошибок не возникло. | Отсутствует |

Для тестирования функционала веб-приложения тест-кейс №2(таблица 4.4 и таблица 4.5) результаты его прохождения приведены в таблице 4.6:

Таблица 4.4. Основная надпись тест-кейса номер 2

|  |  |
| --- | --- |
| **Номер тест-кейса** | 2 |
| **Заголовок** | Осуществить тестирование функционала приложения, взаимодействующего с базой данных. |
| **Предусловие** | Авторизоваться под учетной записью админитратора |

Таблица 4.5. Тест-кейс номер 2

|  |  |  |
| --- | --- | --- |
| **Номер шага** | **Действие** | **Ожидаемый результат** |
| 1 | Нажать кнопку анкетные данные. | Открытие нового раздела, корректное отображение ранее введенной инормации. |
| 2 | Изменить несколько полей в анкете, нажать кнопку сохранить. | Интерактивное уведомление вверху страницы «Данные сохранены» |
| 3 | Открыть вкладку Личные документы. | Открытие нового раздела, корректное отображение ранее введенной инормации. |
| 4 | Загрузить картинку в личные документы, нажав на кнопку «Загрузить» | Оповещение о корректности загрузки «файл загружен», добавление файла в список. |

Таблица 4.5. Тест-кейс номер 2. Продолжение.

|  |  |  |
| --- | --- | --- |
| 5 | Выбрать картинку в списке и нажать кнопку «Установить личное фото» | Оповещение об успешности установки «Успешно утсановлено», замена картинки профиля. |
| 6 | Выбрать не картинку, нажать кнопку «Установить личное фото» | Оповещение об успешности установки «Успешно утсановлено», замена картинки профиля на картинку по умолчанию. |

Таблица 4.6. Выполнение тест-кейса №2

|  |  |  |
| --- | --- | --- |
| **Шаг** | **Примечание** | **Скриншоты** |
| 1 | Выполнено, ошибок не возникло. | Отсутствует |
| 2 | Выполнено, ошибок не возникло. | Отсутствует |
| 3 | Выполнено, ошибок не возникло. | Отсутствует |
| 4 | Выполнено, ошибок не возникло. | Отсутствует |
| 5 | Выполнено, ошибок не возникло. | Отсутствует |
| 6 | Выполнено, ошибок не возникло. | Отсутствует |

Остальные тест-кейсы описываться не будут.

### ЗАКЛЮЧЕНИЕ

В ходе выполнения данной курсовой работы было разработано веб-приложение для автоматизации работы корпуса общественных наблюдателей Российского союза молодежи по Курской облатси.

Работа выполнялась в несколько этапов: была выбрана архитектура, СУБД и разработана структура системы, реализовано веб-приложение.

Веб- приложение разработано, оно успешно функционирует, со своими задачами справляется.

Все цели достигнуты, а задачи выполнены.

### ЛИТЕРАТУРА

1. Официальный сайт Microsoft. Visual Studio Code <https://visualstudio.microsoft.com/ru/>

2. Статья о Visual Studio Code на BizzApps <https://bizzapps.ru/p/vs-code/>

3. Статья об HTML на Википедии <https://ru.wikipedia.org/wiki/HTML>

4. Статья о JavaScript на Википедии <https://ru.wikipedia.org/wiki/JavaScript>

5. Официальный сайт PHP <https://www.php.net/>

6. Статья о PHP на Википедии <https://ru.wikipedia.org/wiki/PHP>

7. Официальный сайт Oracle. Oracle Database 10g XE https://www.oracle.com/ru/database/technologies/appdev/xe.html

8. Шнырёв, С.Л.  Базы данных: учебное пособие для вузов - М. :НИЯУ МИФИ, 2011. — 224 с. — Режим доступа: http://e.lanbook.com/book/75809

9. Ревунков, Г.И. Базы и банки данных - М.: МГТУ им. Н.Э. Баумана, 2011. — 68 с. — Режим доступа: http://e.lanbook.com/book/52425

10. Ревунков, Г.И. Проектирование баз данных - М.: МГТУ им. Н.Э. Баумана, 2009. — 20 с. — Режим доступа: http://e.lanbook.com/book/52390

11. Кудрявцев, К.Я. Создание баз данных: учебное пособие — М.: НИЯУ МИФИ, 2010. — 155 с. — Режим доступа: http://e.lanbook.com/book/75822

12. Сидоров В.Н., Сломинская Е.Н., Полникова Т.В., Макарова О.Ю. Оформление графической части выпускной квалификационной работы. Учебное пособие. М.: МГТУ им. Н.Э. Баумана, 2016.