**ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ И НАУКИ ГОРОДА МОСКВЫ**

**ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ГОРОДА МОСКВЫ**

**«МОСКОВСКИЙ КОЛЛЕДЖ УПРАВЛЕНИЯ, ГОСТИНИЧНОГО БИЗНЕСА И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ «ЦАРИЦЫНО»**

**ДИПЛОМНЫЙ ПРОЕКТ**

Тема: Разработка платформы дистанционного обучения для образовательных учреждений

Пояснительная записка

ДП.09.02.03.13.07.2021.01 ПЗ

Руководитель проекта: Климова Г.Л. Дипломник: Корнев А.Д.

Рецензент: Лисицын И.Н.

**Допущен к защите на заседании**

**Государственной экзаменационной комиссии**

Руководитель учебно-методического

Отдела ОУИТ

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_2021 г. \_\_\_\_\_\_\_\_\_\_ Т.Н. Михайлова

Москва, 2021

|  |  |
| --- | --- |
| **Согласовано**  На заседании Кафедры  информационных технологий  Протокол № 9  от «30» марта 2021 г.  Заведующий кафедрой  \_\_\_\_\_\_\_\_\_\_\_ Е.Ф. Писчасова | **Утверждаю**  Руководитель  учебно-методического  отдела ОУИТ  «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_2021 г.  \_\_\_\_\_\_\_\_\_\_ Т.Н. Михайлова |

**ЗАДАНИЕ НА ДИПЛОМНОЕ ПРОЕКТИРОВАНИЕ**

Студенту Корневу Алексею Дмитриевичу

группы 38ИС-13 специальности 09.02.07 Информационные системы и программирование

апреля

01

Дата выдачи задания: «\_\_» \_\_\_\_\_\_\_\_ 2021 г.

мая

28

Срок сдачи проекта: «\_\_» \_\_\_\_\_\_\_\_\_ 2021 г.

**Тема дипломного проекта:**

Разработка платформы дистанционного обучения для образовательных учреждений

**Техническое задание**

Исходные данные: данные о учебном учреждении, о учебном плане, о расписании занятий, о программных средствах используемых до этого.

**Задачи:**

1. Анализ предметной области
2. Сбор и формализация требований к ПО
3. Разработка структуры ПО и схемы базы данных
4. Разработка алгоритмов решения задачи
5. Разработка модулей ПО
6. Отладка и тестирование ПО
7. Написание руководства пользователя

**Содержание пояснительной записки**

|  |  |
| --- | --- |
| Раздел пояснительной записки | Объем, % |
| Введение | 1 |
| [1 Теоретическая часть](#_Toc11351905)  [1.1 Анализ предметной области](#_Toc11351906)  [1.2 Постановка задачи](#_Toc11351907)  1.3. Обоснование средств разработки | 40 |
| [2 Практическая часть](#_Toc11351913)  [2.1 Разработка базы данных](#_Toc11351914)  [2.2 Работа приложения в виде схем](#_Toc11351915)  [2.3 Разработка интерфейса приложения](#_Toc11351920)  [2.4 Руководство пользователю](#_Toc11351921)  [2.5 Тестирование приложения](#_Toc11351922) | 57 |
| Заключение | 1 |
| Список используемой литературы  Список интернет-источников | 1 |
| Приложение 1. Программный код |  |

**Графическая часть:**

Мультимедиа презентация.

Дипломник: /Корнев А.Д./

Руководитель проекта: /Климова Г.Л./

Заведующий кафедрой: /Писчасова Е.Ф./

Изм.

Лист

№ докум.

Подпись

Дата

Лист

4

**ДП.09.02.03.13.07.2021.01 ПЗ**

Разраб.

Корнев А.Д.

Пров.

Климова Г.Л.

Реценз.

Лисицын И.Н.

*Синицын*

Н. контр.

Климова Г.Л.

Утв.

Писчасова Е.Ф.

Разработка платформы дистанционного обучения для образовательных учреждений

Пояснительная записка

Лит.

Листов

64

**ГБПОУ Колледж «Царицыно»**

Д

СОДЕРЖАНИЕ

[Введение **Ошибка! Закладка не определена.**](#_Toc11351904)

[1 Теоретическая часть **Ошибка! Закладка не определена.**](#_Toc11351905)

[1.1 Анализ предметной области **Ошибка! Закладка не определена.**](#_Toc11351906)

[1.2 Постановка задачи **Ошибка! Закладка не определена.**](#_Toc11351907)

[1.2.1 Функционал приложения Povar-Pro **Ошибка! Закладка не определена.**](#_Toc11351908)

[1.2.2 Требования к входным данным и выдающимся сообщениям **Ошибка! Закладка не определена.**](#_Toc11351909)

[1.3 Средства для реализации **Ошибка! Закладка не определена.**](#_Toc11351910)

[1.3.1 Серверная часть **Ошибка! Закладка не определена.**](#_Toc11351911)

[1.3.2 Клиентская часть. **Ошибка! Закладка не определена.**](#_Toc11351912)

[2 Практическая часть **Ошибка! Закладка не определена.**](#_Toc11351913)

[2.1 Разработка базы данных **Ошибка! Закладка не определена.**](#_Toc11351914)

[2.2 Работа приложения в виде схем **Ошибка! Закладка не определена.**](#_Toc11351915)

[2.3 Разработка интерфейса приложения **Ошибка! Закладка не определена.**](#_Toc11351920)

[2.4 Руководство пользователю **Ошибка! Закладка не определена.**](#_Toc11351921)

[2.5 Тестирование приложения **Ошибка! Закладка не определена.**](#_Toc11351922)

[Заключение **Ошибка! Закладка не определена.**](#_Toc11351923)

[Список используемой литературы **Ошибка! Закладка не определена.**](#_Toc11351924)

[Приложение 1. XAML код **Ошибка! Закладка не определена.**](#_Toc11351925)

[Приложение 2. C# код **Ошибка! Закладка не определена.**](#_Toc11351926)

[Приложение 3. Rest API написанное на PHP **Ошибка! Закладка не определена.**](#_Toc11351927)

# ВВЕДЕНИЕ

Интернет развивается довольно стремительно. Быстро растет количество изданий, посвященных Сети, что предвещает широкое ее распространение даже в далеких от техники областях. Интернет превращается из большой игрушки для интеллектуалов в полноценный источник разнообразной полезной информации для любой категории пользователей.

На данный момент общее количество пользователей всемирной паутины насчитывает 4 500 000 000 человек и с каждым днем эта цифра растёт. Интернет стал неотъемлемой частью каждого пользователя, он заменил библиотеки и специальные справочники. И такая востребованность Сети порождает спрос у компаний к найму специально обученных специалистов – программистов веб-приложений, front-end и back-end разработчиков.

В настоящее время создано большое количество сайтов для любого пользователя: социальные сети, интернет магазины, сайты государственных услуг, онлайн курсы и множество других направлений. За последние пару лет в многих образовательных учреждениях внедряется система онлайн обучения, с помощью специальных площадок на которых выкладывается методический материал, имеется возможность сдавать лабораторные работы, выходить в онлайн конференции для проведения занятий или экзаменов. Эта система имеет множество плюсов, например: не требуется платить за аренду помещения т.к. можно проводить занятия из дома, минимальные системы требования для использования браузера и программного обеспечения видео-конференций, гибкий график занятий.

В связи с событиями весны 2020 года вопрос дистанционного онлайн обучения встал очень остро. Таким образом пришла идея о создании единой платформы дистанционного обучения для образовательных учреждений. Также данная платформа может быть использована не только при особых ситуациях в стране, но и для внедрения направлений дистанционного обучения в очном и заочном формате, обучение специальных групп людей с ограниченными возможностями.

Данный дипломный проект выполнен в соответствии с заданием на дипломное проектирование. Темой проекта является разработка платформы дистанционного обучения для образовательных учреждений.

Целью данного дипломного проекта является оптимизация учебного процесса в период особого положения или по запланированной учебной программе за счет использования специализированной единой платформы дистанционного обучения.

Для достижения поставленной цели необходимо решить ряд задач:

1. Проанализировать предметную область.
2. Проанализировать требования пользователей.
3. Выбрать необходимые технологии и программные средства для реализации.
4. Разработать базу данных.
5. Реализовать необходимый функционал.
6. Реализовать платформу с интуитивно понятным интерфейсом и функциями.

# ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

## Анализ предметной области

В качестве предметной области проекта выступает разработка единой платформы дистанционного обучения. Основной функцией платформы является оптимизация учебного процесса для преподавателей и студентов в период учебного года, особого положения или при установленной учебной программе дистанционного обучения.

Информационный ресурс создается для студентов и сотрудников учебных заведений, чтобы улучшить и оптимизировать работу преподавателей, наладить процесс обучение в дистанционном формате.

Основной упор, создаваемого веб-сайта, направлен на оптимизацию учебного процесса в среднепрофессиональных и высших учебных учреждениях. Платформа позволит студентам продолжить или начать обучение из дома, с минимальными требованиями к используемому носителю, будь то телефон, планшет или персональный компьютер. Преподавателям в свою очередь позволит следить за выполнением учебного плана, получить автоматизированную статистику отработанных часов в каждой группе, выгрузить отчет по «форме №2» за отдельный месяц, получить своевременно информацию о сданных работах и не отвлекаться на посторонние интернет ресурсы, например: электронную почту, гугл класс и т.д.

На данный момент у проведения дистанционного формата обучения есть свои минусы, отсутствует единый ресурс, на котором будет собрана вся информация для обучения, преподавателям и студентам приходится использовать множество сторонних ресурсов, программ, из-за большого объема информации не редко происходит утеря файлов или заданий, что препятствует своевременному освоению учебного плана.

Из этого можно выделить основные недостатки существующей в ГБПОУ Колледж «Царицыно» практики по организации дистанционного обучения:

* Низкая производительность труда из-за большого объема информации;
* Потеря заданий и методических материалов;
* Отсутствует возможность архивировать данные по дистанционному обучению;
* Затруднение с проверкой работ как целых групп, так и отдельных студентов;
* Использование сторонних сервисов, что является не надежным;
* Отсутствие автоматизированной системы для сбора статистики и информации по проведенным занятиям и прочитанным часам;
* Отсутствие единой базы документов.

Для решения выше перечисленных проблем создадим и рассмотрим функциональную диаграмму веб-сайта (Рис. 1)

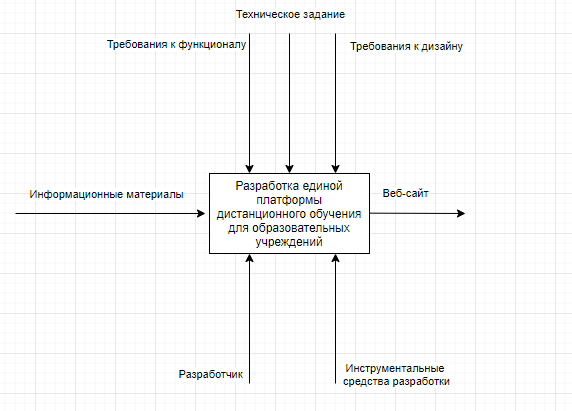


Рис. 1 Функциональная диаграмма

Далее для наглядности, построим декомпозицию решения поставленной задачи (Рис. 2).

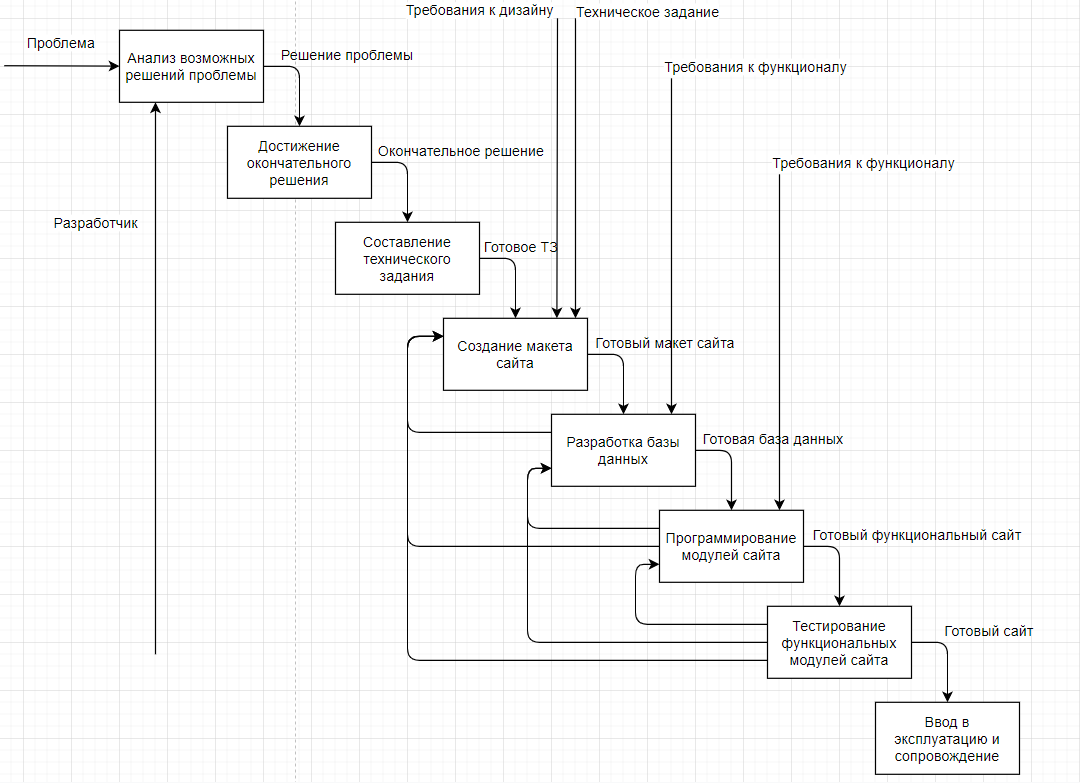


Рис. 2 Декомпозиция решения поставленной задачи

## Постановка задачи

Детально анализируя состояние предметной области и выявив основные недостатки существующей системы, приходим к выводу о целесообразности разработки платформы для оптимизации работы колледжа в условиях дистанционного обучения. На платформе будет отсутствовать регистрация на прямую, для защиты от постороннего вмешательства, доступ и регистрацию будут выдавать специально обученные администраторы платформы от учебного учреждения.

Администраторы смогут регистрировать новых пользователей, добавлять и удалять новые группы студентов, дисциплины, учебную нагрузку, редактировать данные пользователей.

Студенты после получения реквизитов смогут зайти на платформу, ознакомиться со своими оценками, выполненными заданиями, загружать файлы для выполнения заданий.

Преподаватели после получения реквизитов смогут зайти на платформу, выставлять оценки, создавать задания, ознакомиться с статистикой проведенных занятий, выгрузить форму отчета, с данными по часам по выбранному месяцу в формате Excel таблицы (Рис. 3).

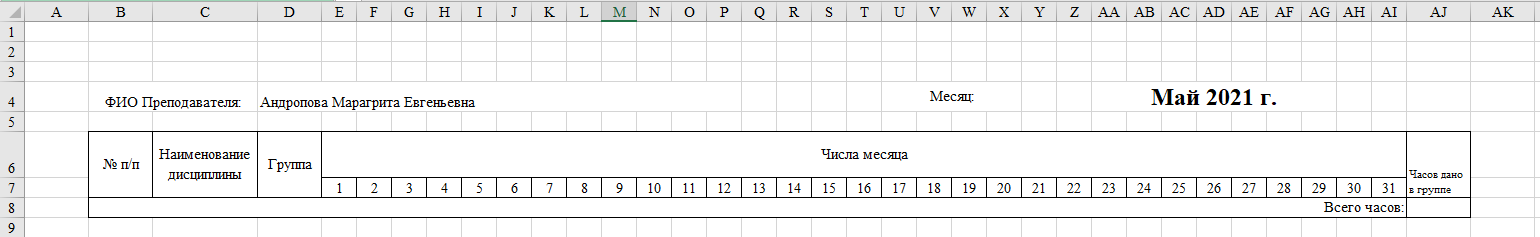


Рис. 3 Форма отчета в формате Excel таблицы.

Для реализации дипломного проекта были поставлены основные задачи по необходимому функционалу.

1. Анализ проблемы и варианты ее решения, составление технического задания, программно-аппаратные решения. Для решения этой задачи необходимо провести анализ проблемы и выбрать наилучший вариант ее решения, составить техническое задание по выбранному решению с включением всех требований по функционалу, интерфейсу и т.д. Выбрать необходимые технические средства для реализации выбранного решения такие как, среда разработки, языки программирования, СУБД, сервер (при необходимости).

2. Автоматизировать регистрацию и авторизацию пользователей. Для решения этой задачи необходимо описать и добавить алгоритм, авторизации и регистрации новых пользователей наиболее подходящий для проекта. Предусмотреть защиту данных от утечки и взлома базы данных.

3. Автоматизировать систему добавления и удаления групп, дисциплин и учебной нагрузки. Для решения этой задачи необходимо описать и добавить алгоритм, перечисленных функций наиболее подходящий для проекта. Протестировать функции на корректную работу.

4. Оптимизировать дистанционное обучение посредством внедрения автоматизированной системы личного кабинета преподавателей и студентов. Для решения этой задачи необходимо узнать и описать весь требуемый функционал для преподавателей и студентов, который необходим для оптимальной работы платформы. Реализовать такие функции как, создание задания, прикрепление файла, выставление оценки за выполненную работу студента.

5. Автоматизированный сбор выполненных работ студентов. Для решения этой задачи необходимо реализовать функцию сбора информации из базы данных и корректное отображение ее для преподавателей и студентов. Отображение будет производится в виде заполняемой шкалы.

6. Автоматизированный сбор статистики для отчетности преподавателей. Для реализации этой задачи необходимо реализовать функцию создания отчета с использованием информации из базы данных, автоматическое заполнение шаблона в Excel таблице с возможностью скачать её.

### Требования к функционалу

На веб-платформе должны быть предусмотрены следующие функции:

* Возможность войти в личный кабинет;
* Возможность зарегистрировать новых пользователей, под учетной записью администратора учебного учреждения;
* Возможность администраторам редактировать профили зарегистрированных пользователей, в случаях утери пароля от учетной записи, смены фамилии, имени или отчества, в случае перевода в другую группу или на другое отделение;
* Возможность администраторам добавлять учебные группы, дисциплины и учебную нагрузку для проведения дистанционных занятий;
* Возможность администраторам вносить изменения в учебную нагрузку при необходимости;
* Двухфакторная защита ввода данных администраторами с помощью пин-кода;
* Возможность заархивировать данные за семестр обучения сроком на 4 года;
* Реализовать вход по ролям, для разделения функционала между, студентами, преподавателями и администраторами;
* Возможность преподавателям создавать задания по практическим и теоретическим занятиям для студентов;
* Возможность преподавателям добавлять методические материалы в удобном для них формате файлов;
* Возможность преподавателей просмотреть выполненные и отправленные работы студентов;
* Возможность оценивать выполненные и отправленные работы студентов;
* Возможность преподавателям просматривать свою статистику отработанных часов в определенной группе за определенный месяц;
* Возможность преподавателям скачать заполненную форму отчета №2 за определенный месяц в формате excel таблицы, с учетом всех проведенных занятий, загруженных на платформу;
* Возможность студентам просматривать необходимый методический материал;
* Возможность студентам добавлять к созданным заданиям выполненные работы в удобном для них формате файлов;
* Возможность студентам просматривать свои оценки;
* Возможность студентам просматривать свою статистику выполненных работ.

При дальнейшей разработке, можно с легкостью доработать старый и добавить новый функционал.

### Требования к интерфейсу

Интерфейс и дизайн веб-платформы должны соблюдать следующие требования.

* Интерфейс веб-платформы должен быть удобным и легким в освоении;
* Должны быть обеспечены ясность и четкость понимания пользователями текстов, кнопок;
* Должны быть использованы спокойные, мягкие цвета, согласно цветам логотипа колледжа;
* Интерфейс должен быть отзывчив для пользователей возрастом от 18 до 65 лет.

### Требования к надежности и безопасности

Веб-платформа должна быть защищена от несанкционированного доступа и попыток взлома, а также весь функционал должен быть протестирован и корректно работать.

* Веб-платформа должна адекватно реагировать на любые действия пользователя, в том числе и на ввод некорректных данных;
* Пользователь своей группы не должен иметь возможности попасть на страницу другой группы пользователей;
* При переходе на каждую страницу должен постоянно проверяться доступ пользователя, если же пользователь не имеет доступа к странице, переотправлять его на главную страницу;
* Должна быть обеспечена безопасность сохраняемых данных о пользователях для авторизации и личные данные пользователей;
* Должна быть обеспечена безопасность базы данных от SQL-инъекций и специальных программ по автоматическому подбору паролей.

### Требования к входным данным

На веб-платформе должны быть предусмотрены следующие входные данные.

Информация о студентах:

* Логин
* Пароль
* Фамилия, имя, отчество (если имеется)
* Группа

Информация о преподавателях:

* Логин
* Пароль
* Фамилия, имя, отчество (если имеется)
* Отделение

Информация о группе:

* Название группы
* Отделение

Информация о дисциплине:

* Отделение
* Название модуля
* Название дисциплины

Информация об учебной нагрузке:

* Отделение
* Дисциплина
* Преподаватель
* Группа
* Количество часов

Информация о задании:

* Название задания
* Описание задания
* Методический материал (файлы с расширением: exe/xls/xlsx/doc/docx/pdf/pptx/mp3/zip/7z/mp4/avi/txt/ai/psd/indd/jpeg/png)

### Требования к выходным данным

Пользователю должны выдаваться следующие выходные данные.

Информация о зарегистрированных преподавателях:

* Отделение
* ФИО
* Логин
* Дата регистрации

Информация о зарегистрированных студентах:

* ФИО
* Логин
* Группа
* Дата регистрации

Информация об учебной нагрузке:

* Отделение
* Наименование дисциплины
* Преподаватель
* Группа

Информация для преподавателей об отработанных часах:

* Дата (месяц, год)
* Группа
* Наименование дисциплины
* Количество часов общее
* Количество часов отработанных

Информация о выполненных работах:

* Выполненное задание (файлы с расширением: exe/xls/xlsx/doc/docx/pdf/pptx/mp3/zip/7z/rar/mp4/avi/txt/ai/psd/indd/jpeg/png)
* Дата загрузки работы студентом
* Оценка

Информация о задании:

* Название задания
* Описание задания
* Методический материал (файлы с расширением: exe/xls/xlsx/doc/docx/pdf/pptx/mp3/zip/7z/rar/mp4/avi/txt/ai/psd/indd/jpeg/png)
* Дата создания задания преподавателем

## Анализ существующих разработок

На данный момент для проведения занятий в дистанционном формате преподаватели и студенты используют сторонние интернет ресурсы, для загрузки методических материалов, необходимых медиа файлов и выполненных работ. Такие ресурсы не очень надежны и требуют двойную работу от преподавателей для хранения информации о проведенных занятиях. Рассмотрим основные используемые интернет ресурсы и сайты.

Google drive – сервис хранения, редактирования и синхронизации файлов разработанный компанией Google (Рис. 4). Основной ресурсы для обмена файлами между преподавателем и студентом. Имеет простой и понятный интерфейс, большие возможности по хранению и редактированию файлов, уровни доступа к файлам, через закрытую ссылку или Google аккаунт. Также имеет ограничение по объему хранимой информации в 15 Гб, увеличение хранилища платное. Все изменения требуют ручного вмешательства пользователя. Для использования Google drive требуется учетная запись в google.

Google Classroom – бесплатный веб-сервис разработанный Google, который призван упростить, создание, распространение и оценку заданий безбумажным способом (Рис. 5). Имеет невзрачный и понятный интерфейс, возможность создавать класс, в котором можно выкладывать задания с оценкой, с сроком сдачи и прикреплять к нему необходимые материалы. Подключение пользователей происходит по специальной ссылке и пин-коду. Хранение файлов осуществляется через Google drive. Для подключения к Google Classroom требуется учетная запись в google.

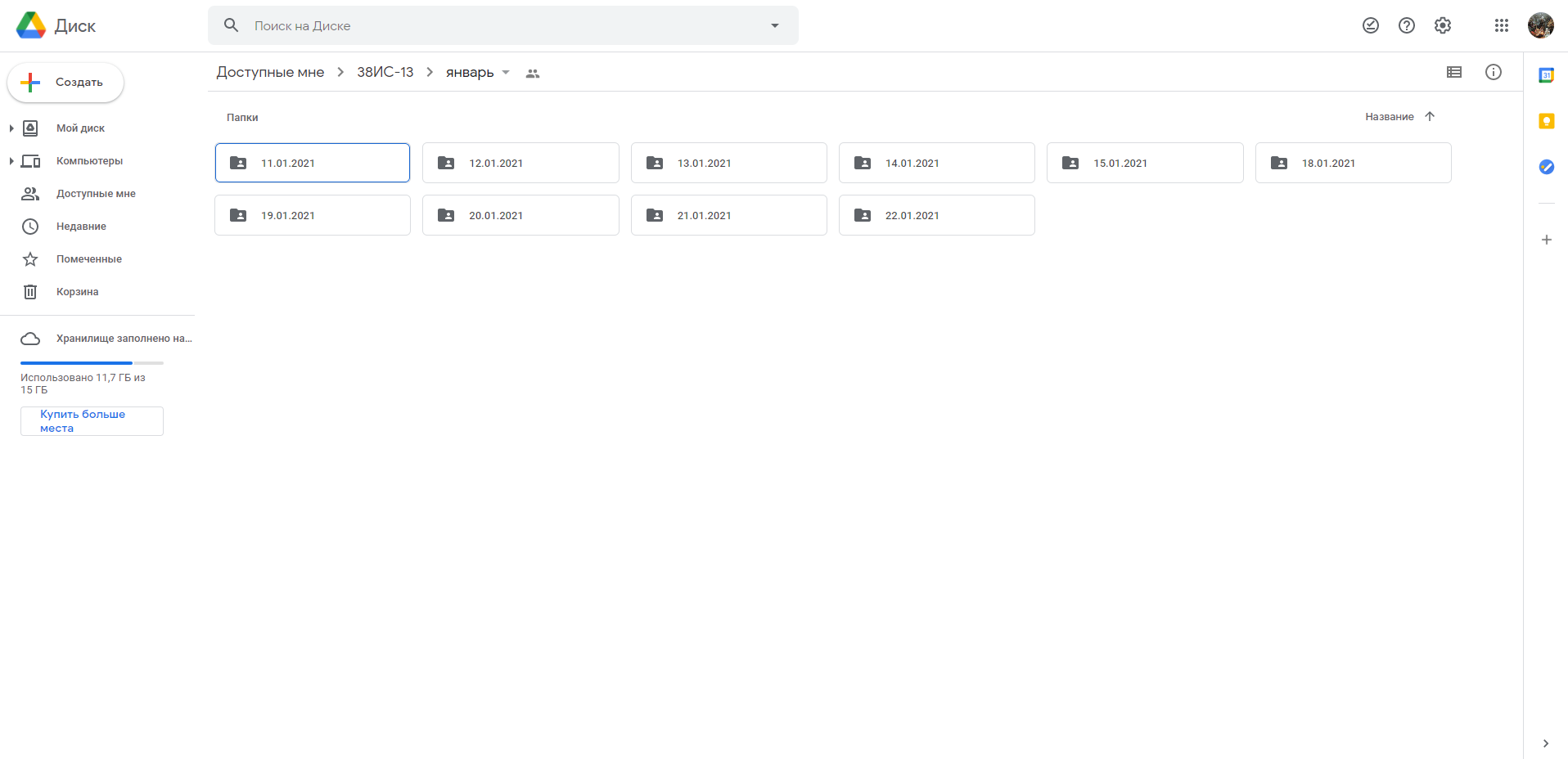


Рис. 4 Интерфейс Google drive

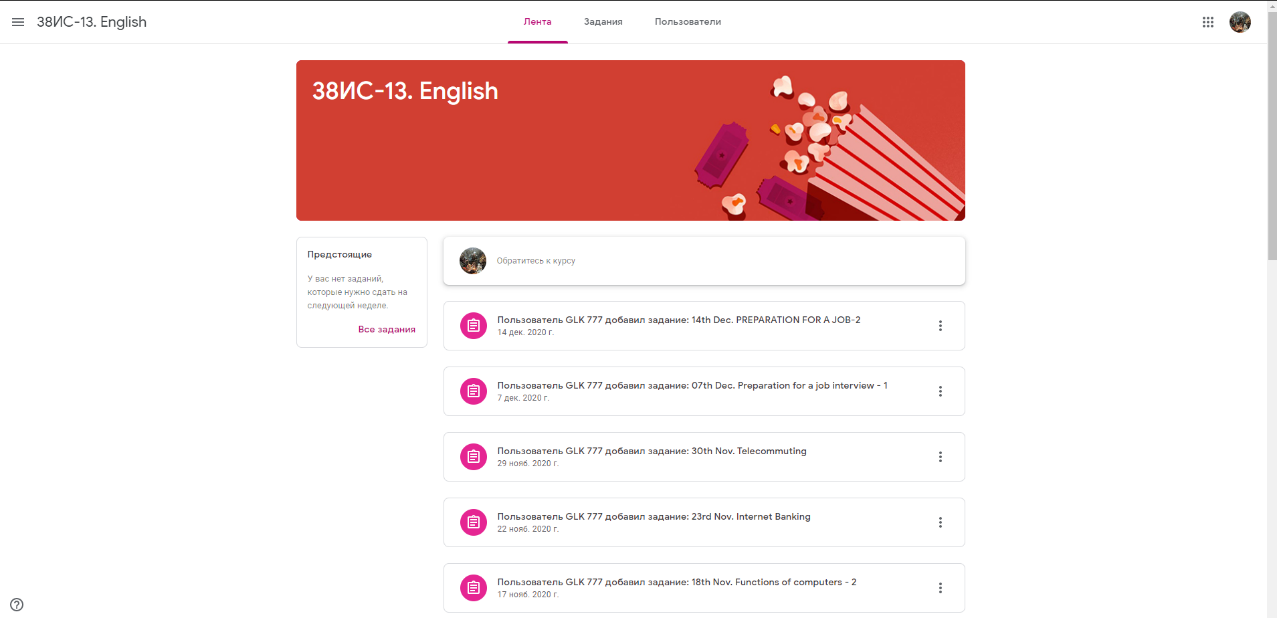


Рис. 5 Интерфейс Google Classroom

Moodle – система управления курсами, также известная как система управления обучением и виртуально-обучающая среда (Рис. 6). Специальное веб-приложение, с помощью которого можно создавать веб-сайты для онлайн обучения. Гибкий дизайн и интерфейс с возможностью конфигурировать макеты и дизайн отдельных страниц. Распространяется по платной подписке, имеет ограничения в количестве подключаемых пользователей, а также ограничения в объеме памяти. Для неопытного пользователя очень сложна в освоении. Есть возможность совместного управления документами. Для использования требуется покупка лицензии на год и аккаунт в Moodle.

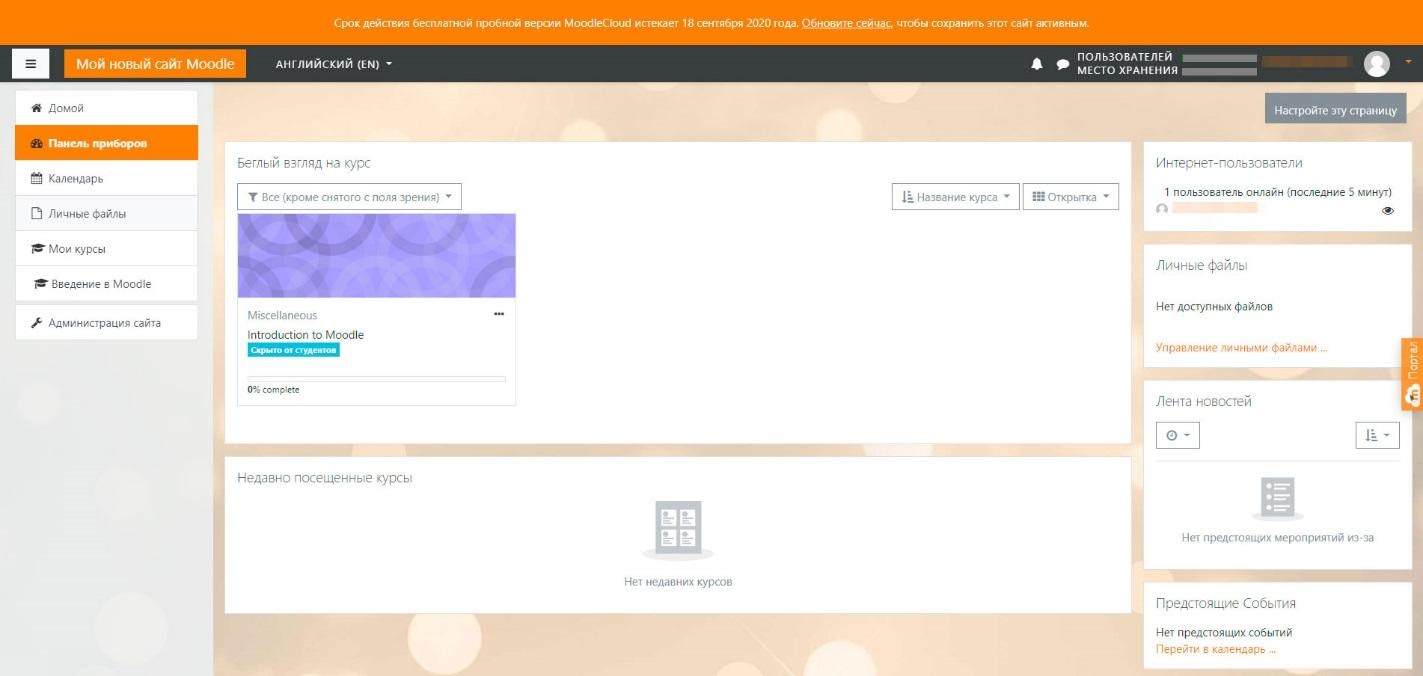


Рис. 6 Интерфейс Moodle

Для наглядности сравним эти информационные системы в таблице 1.

Таблица 1. Сравнение с существующими аналогами

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Наименование информационной системы, сайта | Достоинства | Недостатки |
| 1 | Google drive | Простой и понятный интерфейс;  Возможность загружать файлы в различных форматах и делиться ими;  Бесплатное хранилище на 15 Гб памяти;  Для использования требуется учетная запись google;  Защита от вирусов. | Увеличение хранилища данных платное;  Отсутствует возможность архивации данных;  При утере логина и пароля, данные не восстановить;  Сторонний ресурс. |

Продолжение Таблицы 1.

|  |  |  |  |
| --- | --- | --- | --- |
| 2 | Google Classroom | Невзрачный и интуитивно понятный интерфейс;  Возможность создавать классы для распространения заданий только для приглашенных пользователей;  Система оценок для заданий;  Оповещения на Gmail о изменениях в классе;  Для использования требуется учетная запись google. | На прямую связана с Google drive для хранения файлов;  Отсутствует возможность архивации данных;  Ссылки на подключение бывают битыми;  При утере логина и пароля, данные не восстановить;  Сторонний ресурс. |
| 3 | Moodle | Интерфейс и дизайн можно сделать самому;  Большой функционал для ведения электронного класса;  Защита от вирусов и несанкционированного доступа. | Для использования необходимо покупать лицензию на год;  Очень сложен в освоении;  Документация к использованию платформы на английском языке;  Ограничения в объеме используемой памяти;  Ограничения по одновременно сидящем пользователям на платформе;  Сторонний ресурс. |

В ходе сравнения с другими используемыми ресурсами при дистанционном обучении заметна нехватка единого бесплатного онлайн ресурса, с необходимым функционалом, простым и интуитивно понятным интерфейсом, без ограничений в одновременно подключенных пользователях и объеме памяти. Ввиду того, что данная платформа является дипломным проектом, при доработке определенных модулей, добавления дополнительных необходимых функций, платформу можно вводить в эксплуатацию для дистанционного обучения.

## Обоснование выбора по техническому обеспечению

Техническое обеспечение для реализации проекта требует минимальных затрат:

* Серверное оборудование для установки сайта на домен;
* Процессор: Процессор для серверов INTEL Xeon E3-1230 v6, для ускоренной обработки запросов пользователей, отклика сайта к пользователю, для передачи данных в базу данных;
* Оперативная память в 32 ГБ, для обработки большой информации и передачи ее на сервер или пользователю.

## Обоснование выбора по программному обеспечению

Операционная система при минимальных характеристика компьютера должна поддерживать нынешние браузеры такие как: Google Chrome, Yandex, Safari, Microsoft Edge, Opera.

Для верстки сайта с использование HTML+CSS можно использовать любой текстовый редактор, даже блокнот. Для реализации проекта наилучшим редактором является Visual Studio Code, за счет встроенных плагинов и функций для отладки кода.

Используемые языки программирование это Java Script и PHP v7.2, актуальные языки программирования для использования функций на сайте, связи с базой данных.

СУБД будет использоваться MySQL, это распространенная СУБД используемая почти на каждом сайте, простота в обращении и защищенность, делают ее востребованной.

СУБД MySQL.

MySQL является наиболее приспособленной для применения в среде web СУБД (системой управления базами данных). Не секрет, что для исполнения приложений клиента на большинстве хостинг-площадок провайдеры предоставляют небольшое количество ресурсов (как вычислительных, так и дисковых). Поэтому для данного применения необходима высокоэффективная СУБД, обладающая при этом высокой надежностью (большинство web-приложений и сайтов должны работать в режиме 24/7).

По всем этим причинам MySQL стала незыблемым стандартом в области СУБД для web, а теперь в ней развиваются возможности для использования ее в любых критичных бизнес-приложениях, то есть конкурирует на равных с такими СУБД таких производителей, как Oracle, IBM, Microsoft и Sybase.

Основные преимущества MySQL:

* многопоточность, поддержка нескольких одновременных запросов;
* оптимизация связей с присоединением многих данных за один проход;
* записи фиксированной и переменной длины;
* ODBC драйвер;
* гибкая система привилегий и паролей;
* гибкая поддержка форматов чисел, строк переменной длины и меток времени;
* интерфейс с языками C и Perl, PHP;
* быстрая работа, масштабируемость;
* совместимость с ANSI SQL;
* бесплатна в большинстве случаев;
* хорошая поддержка со стороны провайдеров услуг хостинга;
* быстрая поддержка транзакций через механизм InnoDB.

Язык программирования PHP.

Задачей языка PHP является обеспечение эффективной связи веб-ресурса с сервером и базами данных. При этом данный язык необычайно прост в освоении и применении. По сути, все, что необходимо знать для начала изучения php и программирования на нем, это самый простой язык веб-разметки HTML. PHP прекрасно сочетается с HTML-кодом. Однако для выполнения сценариев одного браузера недостаточно, необходим веб-сервер.

Решаемые задачи.

Код, написанный на PHP, направлен на выполнение двух задач:

* html-часть отвечает за внешний вид и отображение информации;
* php-часть, интегрированная в html, обеспечивает возможности интерактивности и динамику.

При этом подобные программные коды и, соответственно, составленные с их помощью проекты, являются легкими, эффективными, гибкими, многофункциональными, удобными в администрировании, редактировании, обслуживании.

На сегодняшний день подавляющее большинство сайтов, сервисов и приложений, а также такие популярные платформы как Joomla, Drupal, WordPress, 1C-Bitrix и UMI.CMS написаны именно на PHP-языке.

Язык PHP обладает рядом неоспоримых преимуществ:

* Высокая скорость работы и, соответственно, общая производительность ресурсов.
* Бюджетность, экономичность. Найти специалиста не представляется проблемой, стоимость написания программ на php не высока.
* Простота освоения, простой синтаксис.
* Отличная совместимость и переносимость — php-коды работают одинаково хорошо с разными платформами.
* Набор текста кода и его редактирование можно осуществлять в любом текстовом или html-редакторе.
* Высокая гибкость, емкость и функциональность. PHP-программу можно составлять отдельно от разработки веб-страницы, без привязки, после чего совместить. Это существенно упрощает жизнь дизайнеров и программистов.

Многозадачность и широкие возможности — создание любых веб-приложений, блогов, гостевых книг, интернет-магазинов, сайтов, работа с редиректами, заголовками, pdf-документами, базами данных, электронной почтой и пр.

# Практическая часть

## Инициализация проекта

Перед инициализацией проекта необходимо создать план разработки платформы.

* Создать макет визуальной части платформы, подобрать цвета и расположение контейнеров;
* Создать базу данных с необходимыми сущностями и связями между таблицами;
* Сверстать визуальную часть по макету с помощью языков верстки HTML и CSS;
* Реализовать регистрацию новых пользователей через панель администратора и авторизацию пользователей по ролям (студент, преподаватель, администратор);
* Реализовать работу панели администратора, добавить функции добавления групп, дисциплин, учебной нагрузки, предусмотреть возможность редактировать эти данные;
* Реализовать функции создания заданий, оценки заданий и загрузку файлов для преподавателей;
* Реализовать функции загрузки файлов и просмотра загруженных методических материалов для студентов;
* Реализовать функцию отображения отчета о прочитанных часах для преподавателей;
* Реализовать автоматизированную функцию формирование отчета по форме №2.

## Разработка физической и логической структуры сайта

Создадим логическую (Рис. 7) и физическую (Рис. 8) структуру сайта, для наглядности того, как выглядит размещение папок, файлов, стилей, функций и картинок, какое количество этих файлов имеется.

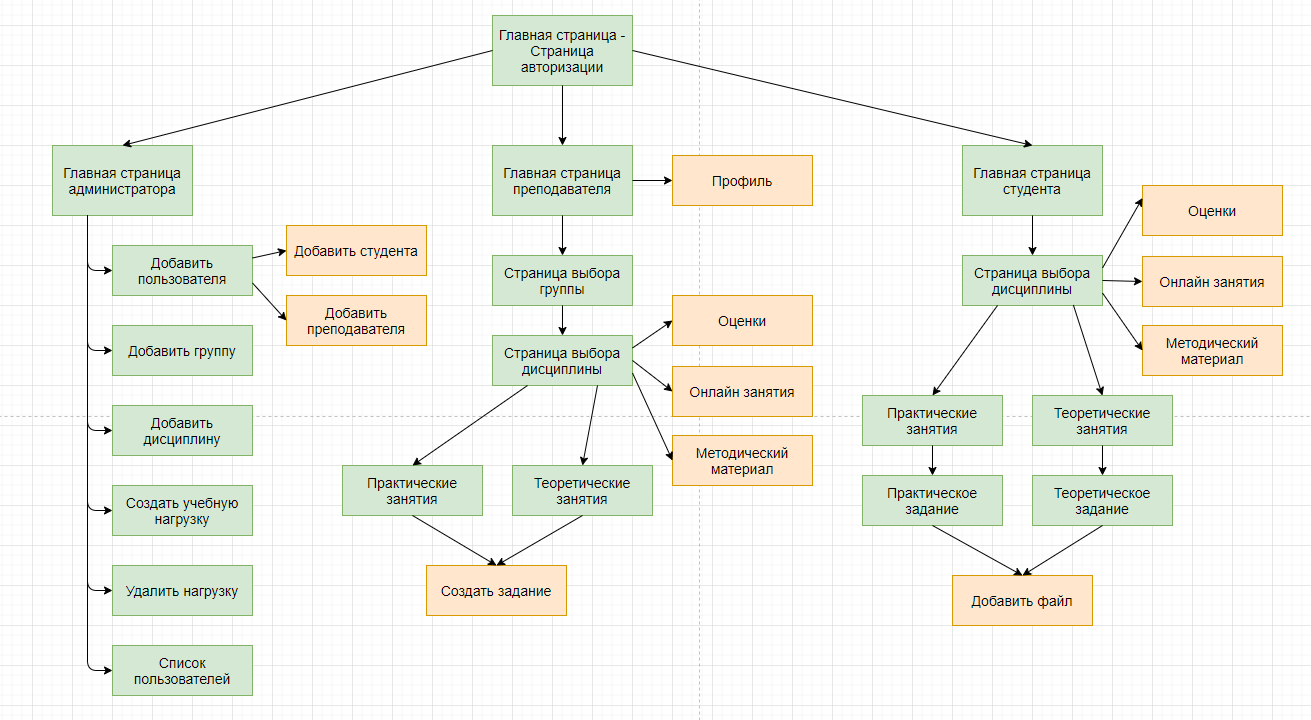


Рис. 7 Логическая структура сайта

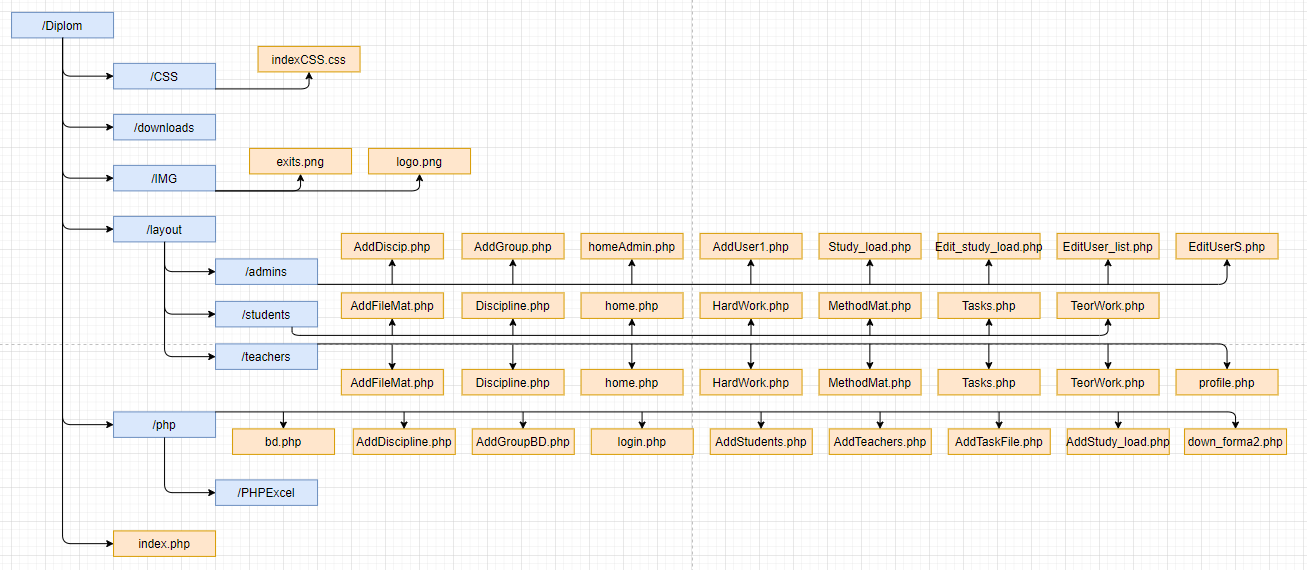


Рис. 8 Физическая структура сайта

## Общие положения (карта сайта)

Создадим визуальную карту сайта, скелет нашего сайта, изображенный в графическом виде, четко структурирован и дает полную информацию о сайте (Рис. 9).

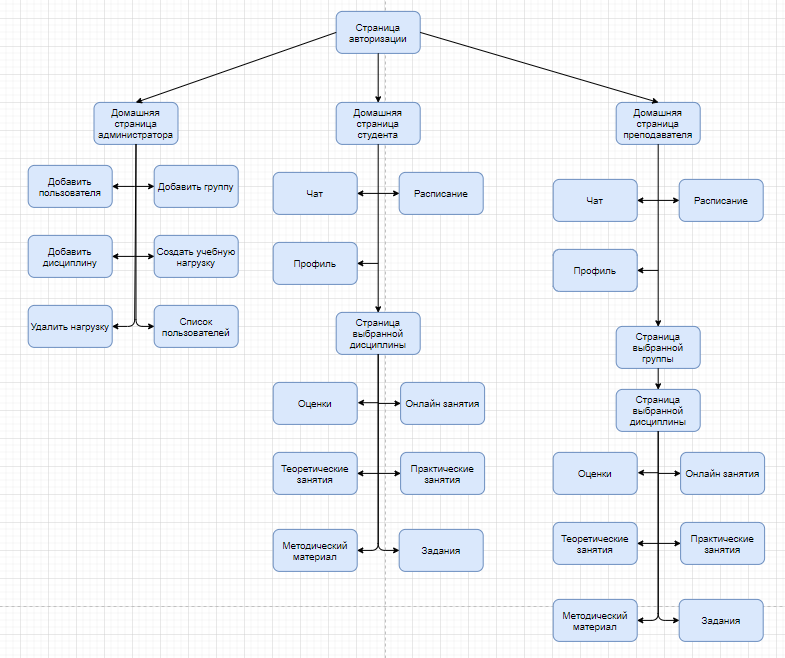


Рис. 9 Карта сайта

## Разработка базы данных

База данных является основным источником получения, хранения, изменения и обмена информацией. Информация в базе данных разделена на таблицы, которые являются сущностями процесса. Для разработки единой платформы дистанционного обучения будет использована СУБД MySQL.

### Выделение множества сущностей

В соответствии с описанием предметной области можно выделить следующие кандидаты в множество сущностей и провести их краткое описание в таблице 2:

Таблица 2. Описание таблиц в базе данных

|  |  |
| --- | --- |
| Название таблицы в базе данных | Описание предназначения таблицы |
| users | Хранит информацию о зарегистрированных пользователях. |
| users\_admins | Хранит информацию об администраторах. |
| users\_students | Хранит информацию о студентах. |
| users\_teachers | Хранит информацию о преподавателях. |
| educational\_inst | Хранит информацию о учебном заведении. |
| corpus | Хранит информацию об отделениях учебных заведениях. |
| discipline | Хранит информацию о дисциплинах. |
| groups | Хранит информацию о группах. |
| study\_load | Хранит информацию об учебной нагрузке. |
| tasks | Хранит информацию о созданных заданиях |
| downloads\_task | Хранит методический материал загруженный к заданиям. |
| hard\_work\_students | Хранит сданные работы студентов по заданиям. |

### Описание множеств сущностей

Описание выделенных множеств сущностей приведено в таблице 3.

Таблица 3. Пул сущностей

|  |  |  |  |
| --- | --- | --- | --- |
| Номер множества сущностей | Имя множества сущностей | Определение множества сущностей | Описание множества сущностей |
| E1 | users | Зарегистрированные пользователи (студенты, преподаватели, администраторы). | Как только регистрируется новый пользователь, формируется новый экземпляр данного множества сущностей |

Продолжение Таблицы 3.

|  |  |  |  |
| --- | --- | --- | --- |
| E2 | users\_admins | Необходимая информация о зарегистрированных администраторах. | Как только регистрируется новый пользователь с правами администратора, формируется новый экземпляр данного множества сущностей. |
| E3 | users\_students | Необходимая информация о зарегистрированных студентах. | Как только регистрируется новый пользователь с правами студента, формируется новый экземпляр данного множества сущностей. |
| E4 | users\_teachers | Необходимая информация о зарегистрированных преподавателях. | Как только регистрируется новый пользователь с правами преподавателя, формируется новый экземпляр данного множества сущностей. |
| E5 | educational\_inst | Информация о учебных заведениях использующих платформу | Как только учебное заведение решает использовать платформу, формируется новый экземпляр данного множества сущностей. |
| Е6 | corpus | Информация о отделениях или филиалах учебных заведений. | Как только учебное заведение регистрируется на платформе, формируется новый экземпляр данного множества сущностей. |

Продолжение таблицы 3.

|  |  |  |  |
| --- | --- | --- | --- |
| Е7 | discipline | Информация о модулях и дисциплинах добавленных администраторами учебных заведений. | Как только администратор добавляет новую дисциплину, формируется новый экземпляр множества сущностей. |
| Е8 | group | Информация о группах добавленных администраторами учебных заведений. | Как только администратор добавляет новую группу, формируется новый экземпляр множества сущностей. |
| E9 | study\_load | Информация о учебной нагрузке добавленной администратором. | Как только администратор добавляет новую учебную нагрузку, формируется новый экземпляр множества сущностей. |
| E10 | tasks | Информация о созданных заданиях преподавателями. | Как только преподаватель создает новое задание, формируется новый экземпляр множества сущностей. |
| E11 | downloads\_task | Информация о загруженном преподавателями методическом материале для заданий. | Как только преподаватель загружает методический материал к заданию, формируется новый экземпляр множества сущностей. |
| E12 | hard\_work\_students | Информация о загруженных студентами работах по заданиям. | Как только студент загружает работу к заданию, формируется новый экземпляр множества сущностей. |

## Построение модели уровня сущностей

### Матрица связей

На основании анализа постановки задачи устанавливаются связи между множествами сущностей в таблице 4.

Таблица 4. Матрица связей

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | E1 | E2 | E3 | E4 | E5 | E6 | E7 | E8 | E9 | E10 | E11 | E12 |
| users/E1 |  | X | X | X |  |  |  |  |  |  |  |  |
| users\_admins/E2 | X |  |  |  | X | X |  |  |  |  |  |  |
| users\_students/E3 | X |  |  |  | X | X |  | X |  |  |  | X |
| users\_teachers/E4 | X |  |  |  | X | X |  |  | X |  |  |  |
| educational\_inst  /E5 |  | X | X | X |  | X |  |  |  |  |  |  |
| corpus/E6 |  | X | X | X | X |  | X | X | X |  |  |  |
| discipline/E7 |  |  |  |  |  | X |  |  | X |  |  |  |
| groups/E8 |  |  | X |  |  | X |  |  | X |  |  |  |
| study\_load/E9 |  |  |  | X |  | X | X | X |  | X |  |  |
| tasks/E10 |  |  |  |  |  |  |  |  | X |  | X | X |
| downloads\_task/E11 |  |  |  |  |  |  |  |  |  | X |  |  |
| hard\_work\_students  /E12 |  |  | X |  |  |  |  |  |  | X |  |  |

### Описание связей

Описание выделенных связей приводится в таблице 5.

Таблица 5. Описание связей

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Номер связи | Номер 1-й сущности | Номер 2-й сущности | Имя связи | Тип связи | Описание связи |
| R1 | E1 родительская | E2 дочерняя | Имеет | Один к одному, обязательная | Каждый пользователь с правами администратора имеет одну запись о информации администратора. Каждая информация о администраторе имеет только одного пользователя. |

Продолжение таблицы 5.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| R2 | E1 родительская | E3 дочерняя | Имеет | Один к одному, обязательная | Каждый пользователь с правами студента имеет одну запись о информации студента. Каждая информация о студенте имеет только одного пользователя. |
| R3 | E1 родительская | E4 дочерняя | Имеет | Один к одному, обязательная | Каждый пользователь с правами преподавателя имеет одну запись о информации преподавателя. Каждая информация о преподавателе имеет только одного пользователя. |
| R4 | E5 родительская | E2 дочерняя | Имеет | Один к многим, обязательная | Каждое учебное заведение имеет множество своих администраторов. Каждый администратор имеет только одно учебное заведение. |
| R5 | E5 родительская | E3 дочерняя | Имеет | Один к многим, обязательная | Каждое учебное заведение имеет множество своих студентов. Каждый студент имеет только одно учебное заведение. |
| R6 | E5 родительская | E4 дочерняя | Имеет | Один к многим, обязательная | Каждое учебное заведение имеет множество своих преподавателей. Каждый преподаватель имеет только одно учебное заведение. |
| R7 | E6 родительская | E2 дочерняя | Имеет | Один к одному, обязательная | Каждое отделение имеет одного своего администратора. Каждый администратор имеет только одно отделение. |

Продолжение таблицы 5.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| R8 | E6 родительская | E3 дочерняя | Имеет | Один к многим, обязательная | Каждое отделение имеет множество своих преподавателей. Каждый преподаватель имеет только одно отделение. |
| R8 | E6 родительская | E4 дочерняя | Имеет | Один к многим, обязательная | Каждое отделение имеет множество своих студентов. Каждый студент имеет только одно отделение. |
| R9 | E5 родительская | E6 дочерняя | Имеет | Один к многим, обязательная | Каждое учебное заведение имеет одно или более отделений. Каждое отделение имеет только одно учебное заведение. |
| R10 | E6 родительская | E7 дочерняя | Имеет | Один к многим, обязательная | Каждое отделение имеет множество дисциплин и модулей. Каждая дисциплина или модуль имеет только одно отделение. |
| R11 | E6 родительская | E8 дочерняя | Имеет | Один к многим, обязательная | Каждое отделение имеет множество учебных групп. Каждая учебная группа имеет только одно отделение. |
| R12 | E7 родительская | E9 дочерняя | Имеет | Один к многим, обязательная | Каждая дисциплина имеет множество учебных нагрузок. Каждая учебная нагрузка имеет только одну дисциплину. |
| R13 | E8 родительская | E3 дочерняя | Имеет | Один к многим, обязательная | Каждая группа имеет множество студентов. Каждый студент имеет только одну группу. |

Продолжение таблицы 5.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| R14 | E8 родительская | E9 дочерняя | Имеет | Один к многим, обязательная | Каждая группа имеет одну или более учебных нагрузок. Каждая учебная нагрузка имеет одну группу. |
| R15 | E4 родительская | E9 дочерняя | Имеет | Один к многим, обязательная | Каждый преподаватель имеет одну или более учебных нагрузок. Каждая учебная нагрузка имеет одного преподавателя. |
| R16 | E6 родительская | E9 дочерняя | Имеет | Один к многим, не обязательная | Каждое отделение имеет одно и более учебных нагрузок. Каждая учебная нагрузка имеет одно отделение. |
| R17 | E9 родительская | E10 дочерняя | Имеет | Один к многим, не обязательная | Каждая учебная нагрузка имеет одно или более заданий. Каждое задание имеет одну учебную нагрузку. |
| R18 | E10 родительская | E11 дочерняя | Имеет | Один к многим, не обязательная | Каждое задание имеет один или более методических материалов. Каждый методический материал имеет одно задание. |
| R19 | E10 родительская | E12 дочерняя | Имеет | Один к многим, не обязательная | Каждое задание имеет множество выполненных работ студентов. Каждая выполненная работа студента имеет одно задание. |
| R20 | E3 родительская | E12 дочерняя | Имеет | Один к многим, не обязательная | Каждый студент имеет множество выполненных работ. Каждая выполненная работа имеет одного студента. |

### Концептуальная схема уровня сущностей

В соответствии с определенными множествами сущностей и связями между ними строится концептуальная схема уровня сущностей (Рис. 10).

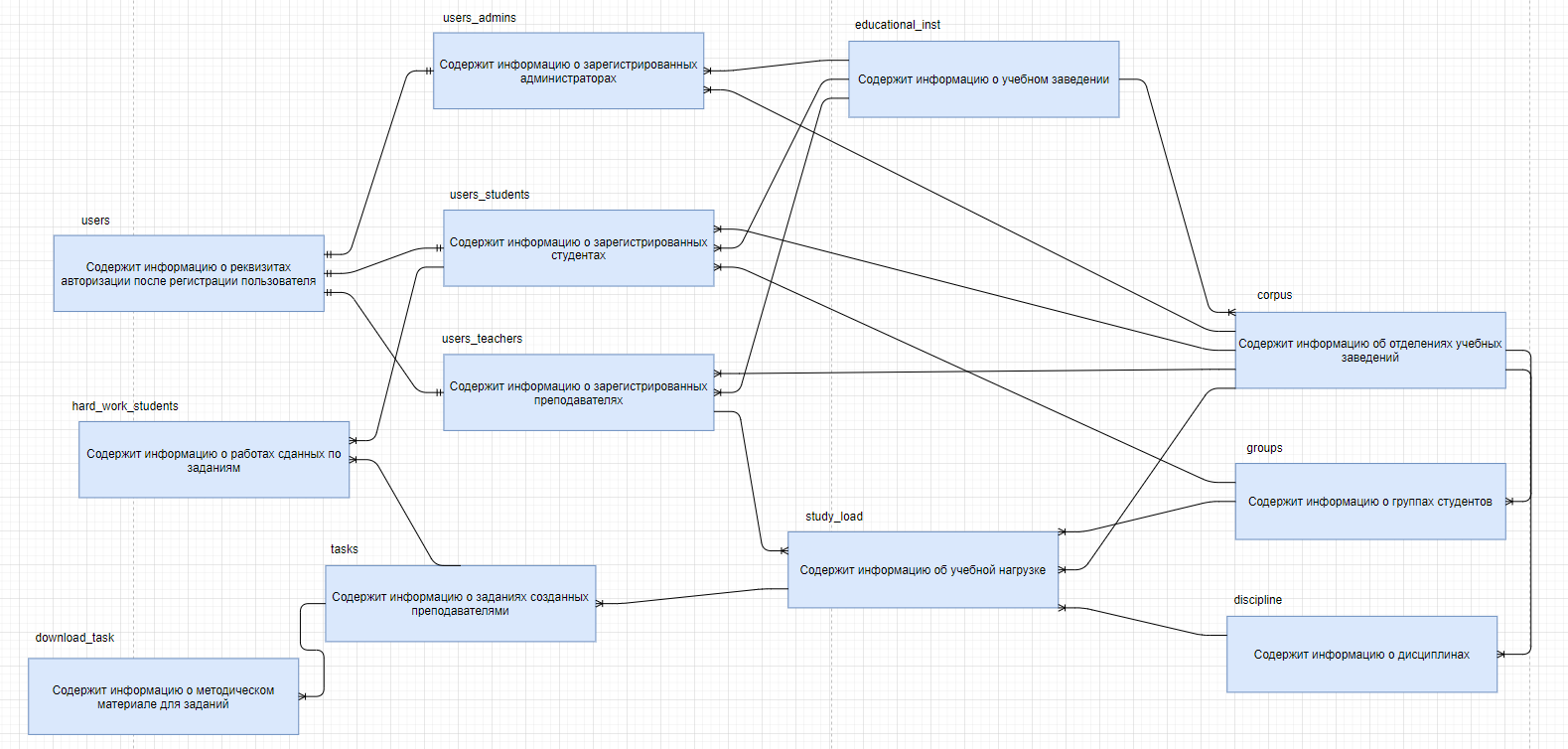


Рис. 10 Концептуальная схема уровня сущностей

## Построение модели уровня ключей

### Определение ключевых атрибутов

Для всех сущностей в качестве первичного ключа использовался искусственный атрибут – ЛИЧНЫЙ НОМЕР (КОД). Значения атрибута – целые положительные числа (порядковый номер).

### Описание атрибутов уровня ключей

Описание атрибутов уровня ключей приводится в таблице 6.

Таблица 6. Описание атрибутов уровня ключей

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Множество сущностей | Имя атрибута | Имя домена | Признак обязательности | Примечание |
| users/E1 | users\_id | Номер | Not null | PK |
| users\_name | Строка | Not null | PK |
| users\_admins/E2 | admins\_id | Номер | Not null | PK |
| admins\_log | Строка | Not null | FK |
| admins\_uz | Строка | Not null | FK |

Продолжение таблицы 6.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| users\_students/E3 | students\_id | Номер | Not null | PK |
| students\_log | Строка | Not null | FK |
| students\_group | Строка | Not null | FK |
| students\_uz | Строка | Not null | FK |
| users\_teachers/E4 | teachers\_id | Номер | Not null | PK |
| teachers\_log | Строка | Not null | FK |
| teachers\_otdel | Строка | Not null | FK |
| teachers\_uz | Строка | Not null | FK |
| educational\_inst/E5 | id\_EI | Номер | Not null | PK |
| name\_EI | Строка | Not null | PK |
| corpus/E6 | id\_corpus | Номер | Not null | PK |
| name\_corpus | Строка | Not null | PK |
| name\_uz | Строка | Not null | FK |
| discipline/E7 | id\_dis | Номер | Not null | PK |
| name\_dis | Строка | Not null | PK |
| name\_corpus | Строка | Not null | FK |
| groups/E8 | id\_groups | Номер | Not null | PK |
| name\_group | Строка | Not null | PK |
| name\_corpus | Строка | Not null | FK |
| study\_load/E9 | id\_SL | Номер | Not null | PK |
| name\_dis | Строка | Not null | FK |
| name\_group | Строка | Not null | FK |
| name\_teacher | Строка | Not null | FK |
| name\_corpus | Строка | Not null | FK |
| tasks/E10 | id\_task | Номер | Not null | PK |
| id\_SL | Номер | Not null | FK |
| downloads\_task/E11 | id\_DT | Номер | Not null | PK |
| id\_task | Номер | Not null | FK |
| hard\_work\_students/E12 | id\_HW | Номер | Not null | PK |
| name\_students | Строка | Not null | FK |
| id\_task | Номер | Not null | FK |

### Концептуальная схема уровня ключей

По результатам определения первичных ключей строится концептуальная схема уровня ключей (Рис. 11).

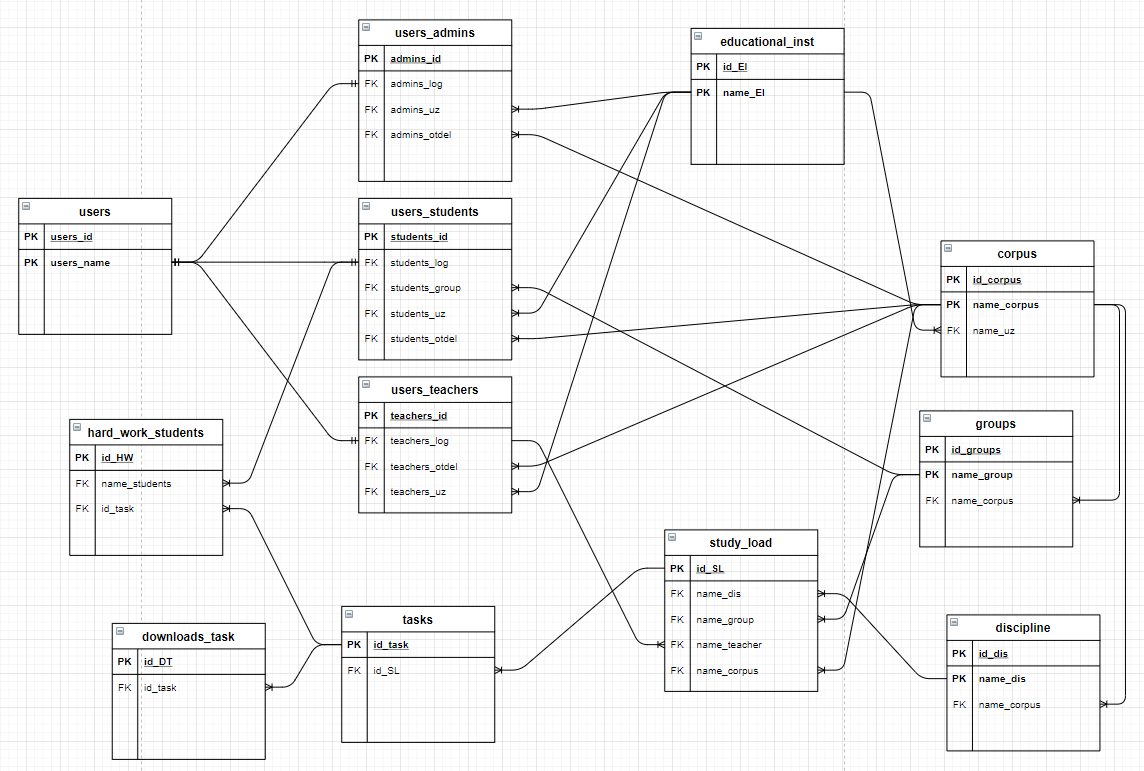


Рис. 11 Концептуальная схема уровня ключей

## Построение полноатрибутной модели

На этом этапе в соответствии с проведенным анализом предметной области определяются все необходимые атрибуты. Описание всех атрибутов и ключей проводится в таблице 7.

Таблица 7. Описание полноатрибутная модели

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Множество сущностей | Имя атрибута | Имя домена | Признак обязательности | Примечание |
| users/E1 | users\_id | Номер | Not null | PK |
| users\_name | Строка | Not null | PK |
| users\_pass | Строка | Not null | Пароль пользователя |
| users\_access | Номер | Not null | Роль на платформе |
| users\_date | Дата | Not null | Дата регистрации |

Продолжение таблицы 7.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| users\_admins/E2 | admins\_id | Номер | Not null | PK |
| admins\_log | Строка | Not null | FK |
| admins\_uz | Строка | Not null | FK |
| admins\_f | Строка | Not null | Фамилия администратора |
| admins\_n | Строка | Not null | Имя администратора |
| admins\_o | Строка | Not null | Отчество администратора |
| admins\_pin | Строка | Not null | Пин-код администратора |
| users\_students/E3 | students\_id | Номер | Not null | PK |
| students\_log | Строка | Not null | FK |
| students\_group | Строка | Not null | FK |
| students\_uz | Строка | Not null | FK |
| students\_f | Строка | Not null | Фамилия студента |
| students\_n | Строка | Not null | Имя студента |
| students\_o | Строка | Not null | Отчество студента |
| users\_teachers/E4 | teachers\_id | Номер | Not null | PK |
| teachers\_log | Строка | Not null | FK |
| teachers\_otdel | Строка | Not null | FK |
| teachers\_uz | Строка | Not null | FK |
| teachers\_f | Строка | Not null | Фамилия преподавателя |
| teachers\_n | Строка | Not null | Имя преподавателя |
| teachers\_o | Строка | Not null | Отчество преподавателя |
| educational\_inst/E5 | id\_EI | Номер | Not null | PK |
| name\_EI | Строка | Not null | PK |
| corpus/E6 | id\_corpus | Номер | Not null | PK |
| name\_corpus | Строка | Not null | PK |
| name\_uz | Строка | Not null | FK |
| discipline/E7 | id\_dis | Номер | Not null | PK |
| name\_dis | Строка | Not null | PK |
| name\_corpus | Строка | Not null | FK |
| name\_modul | Строка | Not null | Название модуля учебных дисциплин |
| groups/E8 | id\_groups | Номер | Not null | PK |
| name\_group | Строка | Not null | PK |
| name\_corpus | Строка | Not null | FK |
| study\_load/E9 | id\_SL | Номер | Not null | PK |
| name\_dis | Строка | Not null | FK |

Продолжение таблицы 7.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | name\_group | Строка | Not null | FK |
| name\_teacher | Строка | Not null | FK |
| name\_corpus | Строка | Not null | FK |
| time\_study\_load | Номер | Not null | Количество часов в нагрузке |
| url\_SL | Длинный текст | Not null | Ссылка на google таблицу закрепленную за учебной нагрузкой |
| url\_edit | Длинный текст | Not null | Ссылка на редактирование google таблицы закрепленной за учебной нагрузкой |
| tasks/E10 | id\_task | Номер | Not null | PK |
| id\_SL | Номер | Not null | FK |
| name\_tasks | Строка | Not null | Название задания |
| description\_tasks | Длинный текст | Not null | Описание задания |
| data\_create | Дата | Not null | Дата создания задания |
| catt\_tasks | Строка | Not null | Категория задания – (практическое или теоретическое) |
| time\_tasks | Номер | Not null | Каждое созданное задание имеет фиксированное значение прочитанных часов |
| downloads\_task/E11 | id\_DT | Номер | Not null | PK |
| id\_task | Номер | Not null | FK |
| path\_DT | Строка | Not null | Путь к файлу методического материала на сервере |
| name\_file | Строка | Not null | Имя файла методического материала |
| date\_create\_DT | Дата | Not null | Дата загрузки методического материала |

Продолжение таблицы 7.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| hard\_work\_students/E12 | id\_HW | Номер | Not null | PK |
| name\_students | Строка | Not null | FK |
| id\_task | Номер | Not null | FK |
| path\_HW | Строка | Not null | Путь к файлу выполненной работы на сервере |
| date\_create\_HW | Дата | Not null | Дата загрузки выполненной работы |
| name\_file | Строка | Not null | Название файла выполненной работы |
| score\_HW | Строка | Null | Оценка за выполненную работу поставленная преподавателем |

На основе этой таблица мы можем спроектировать полноатрибутную концептуальную схему (Рис. 12).

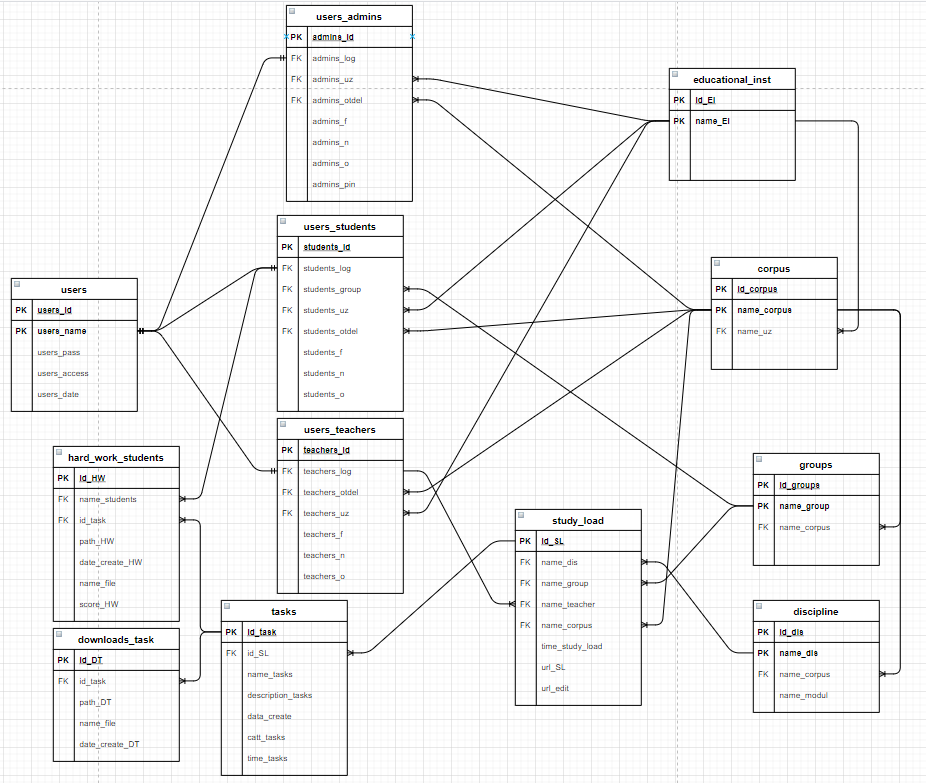


Рис. 12 Полноатрибутная концептуальная схема

## Создание дизайн-макета платформы

Наиболее простым отображением будущего сайта считается его макетная схема. Макет сайта — нечто среднее между блочной схемой и прототипом. Это оптимальный вариант для клиента, которому не придется переплачивать за интерактивные эффекты прототипа. С другой стороны, макет не выглядит слишком простым как блочная схема. С его помощью верстальщику проще понять, каким именно должен быть сайт. На основе этого представим макет страницы авторизации (Рис. 13), главную страницу для преподавателей и студентов (Рис. 14) и главную страницу администратора разрабатываемой платформы (Рис. 15).

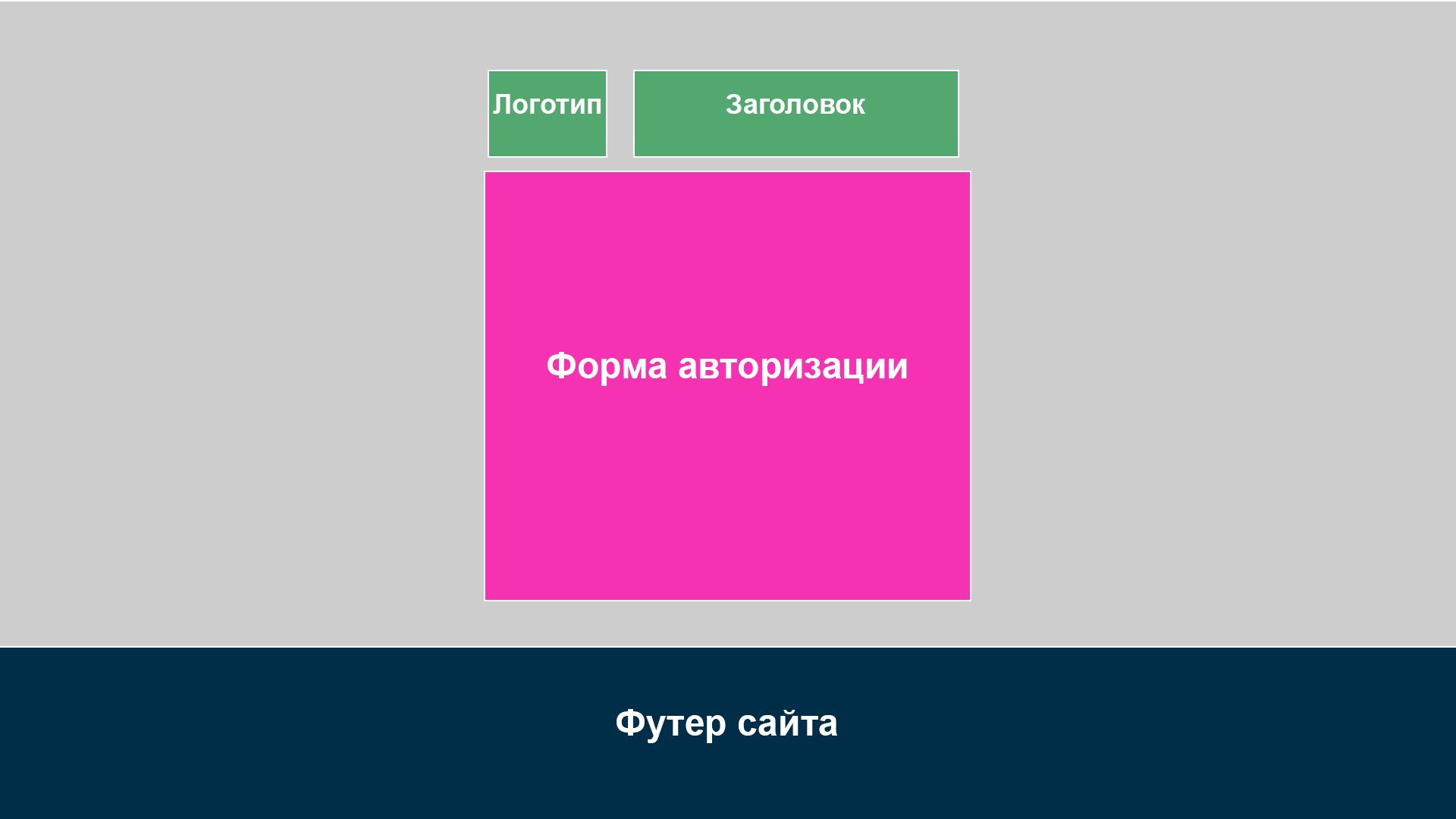


Рис. 13 Дизайн-макет страницы авторизации

На данном дизайн-макете представлено расположение логотипа, заголовка (название платформы), формы авторизации и футер. Дизайн футера будет на всех страницах платформы одинаковый.

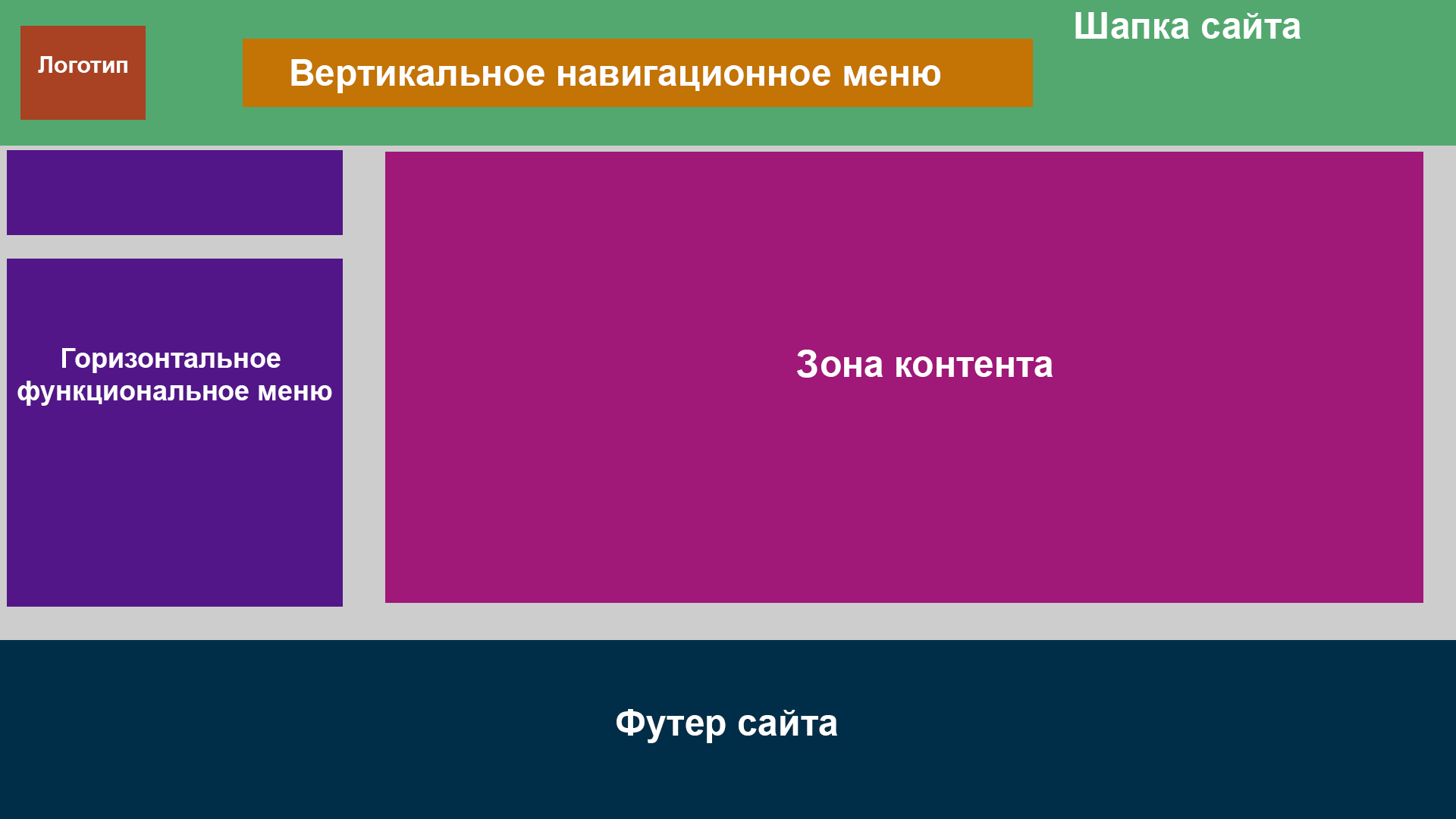


Рис. 14 Дизайн-макет главной страницы преподавателей и студентов

На данном дизайн-макете представлено расположение шапки сайта, логотипа, навигационного меню, функционального меню, зоны контента и футер сайта. Дизайн шапки и футера будет на всех страницах платформы одинаковый.

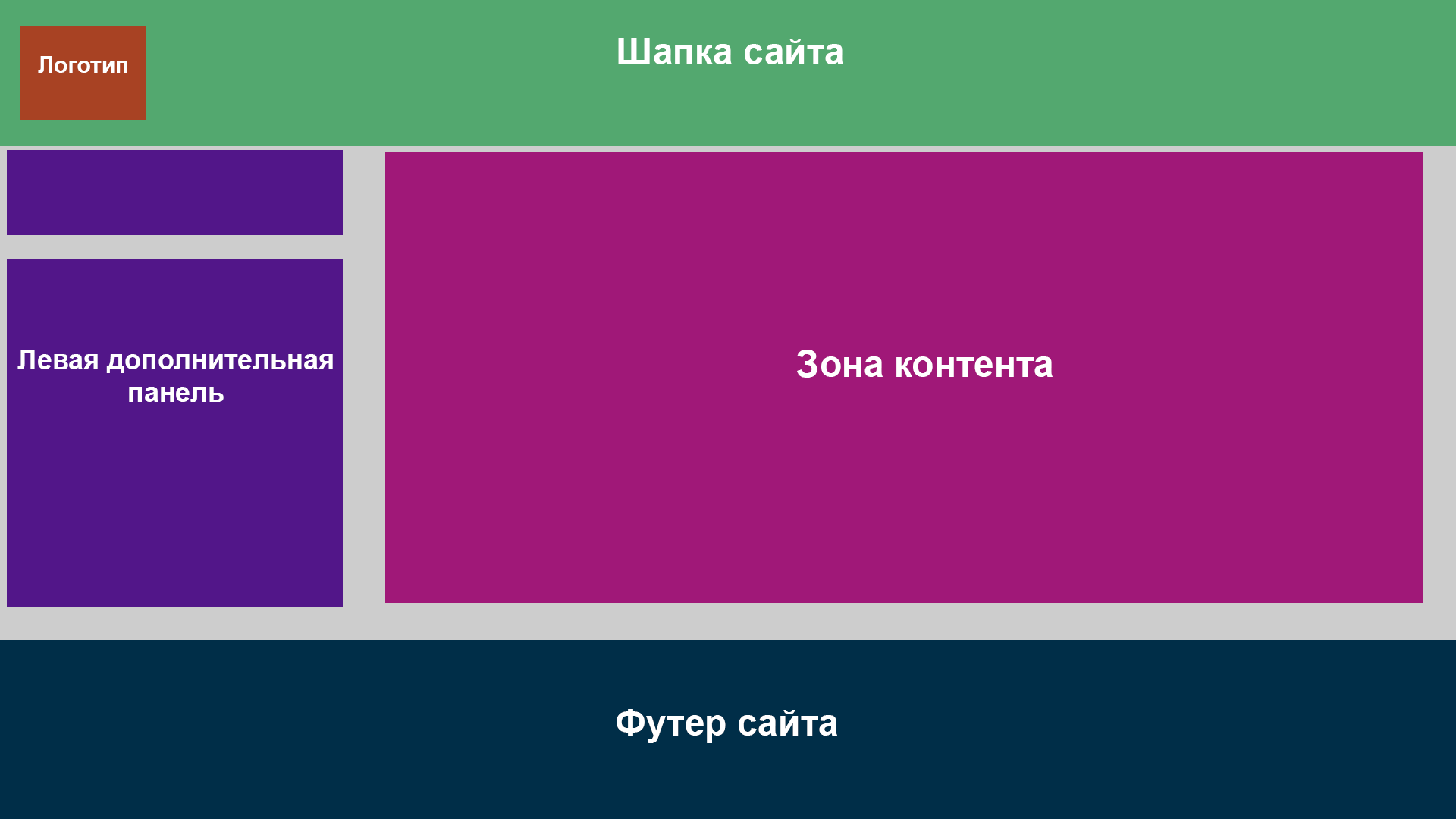


Рис.15 Дизайн-макет главной страницы

На данном дизайн-макете представлено расположение шапки сайта, логотипа, левой дополнительной панели, зоны контента и футер сайта. Навигационная и функциональная панели перемещены в левую часть страницы. Дизайн шапки и футера будет на всех страницах администратора платформы одинаковый.

## Верстка страниц

Верстка html-страниц сайта на основе утвержденного дизайна типовых страниц. Для дизайна страниц используется язык верстки HTML с использованием каскадных таблиц стилей (CSS). Верстка платформы была разделена на несколько этапов: разработку шапки сайта(header), разработку подвала сайта(footer), разработка основной зоны контента(main). Шапка на страницах доступных для преподавателей и студентов не отличается (Рис. 16), в отличии от шапки на страницах доступных администратору (Рис.17). Код верстки шапки представлен в приложении 1.

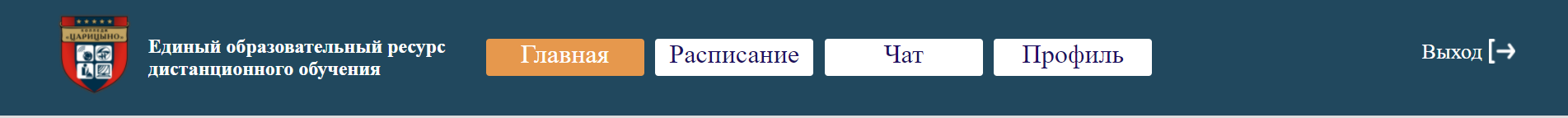


Рис. 16 Визуальное изображение шапки на страницах студентов и преподавателей

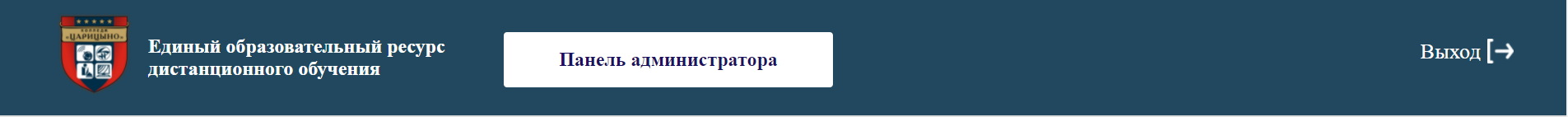


Рис. 17 Визуальное изображение шапки на страницах администраторов

Для дизайна стилей каждого элемента на странице используется CSS (каскадные таблицы стилей) (Рис. 18). Для каждого элемента были использованы классы и селекторы с помощью которых можно настроить всё, начиная от цвета и размера заканчивая обводкой, фоном и размещением. Для ссылок были использованы псевдоклассы с помощью которых можно видоизменять элемент при нажатии или наведении на него. Код каскадных таблиц стилей представлен в приложении 2.

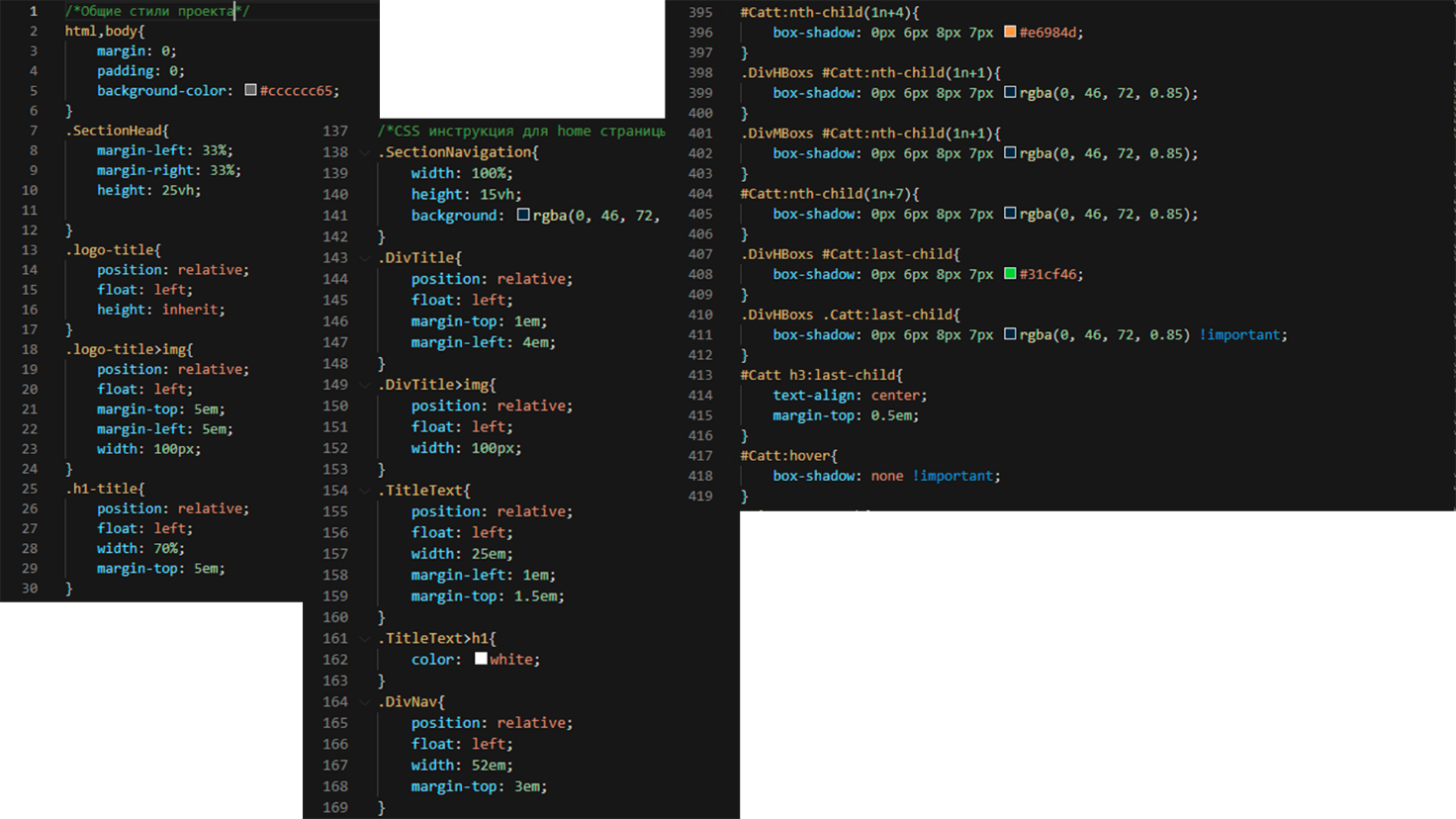


Рис. 18 Использованные методы каскадных таблиц стилей

## Логика работы программы

Для реализации многих функций сайта используется модульное программирование. Как правило, сначала создается главная страница, а затем добавляются все разделы и категории в соответствии с иерархией веб-сайта, которая была ранее создана в виде карты сайта. Все статические элементы веб-страницы, которые были разработаны во время создания макета. Затем следует добавить специальные функции.

Для начала это функционал администратора, функция регистрации новых пользователей и отображения всех зарегистрированных пользователей с возможностью редактировать их профили, добавления новых учебных групп, дисциплин, учебной нагрузки, отображения учебной нагрузки и возможность редактирования этой таблицы.

Далее добавляется функционал преподавателя, функция отображение групп и дисциплин, которые были добавлены в учебную нагрузку администратором для определенного преподавателя, просмотра и редактирование привязанной гугл-таблицы с оценками определенной группы, добавления задания и просмотра уже созданных заданий, загрузки методического материала, отображения студентов и их загруженных работ по заданиям, отображения статистики проведенных занятий и автоматизированного заполнения и загрузки отчета по форме №2.

В конце добавляется функционал студента, функция отображения дисциплин, которые были добавлены в учебную нагрузку администратором для группы в которой обучается студент, просмотра привязанной гугл-таблицы с оценками группы, отображения созданных заданий преподавателями, загрузки своих выполненных работ к этим заданиям, отображения информации о количестве сданных работ, отображения и скачивания методического материала, загруженного преподавателями. Код функций представлен в приложении 3.

Платформа дистанционного обучения тесно работает с базой данных поэтому большая часть функций имеет SQL-запросы. Благодаря им данные вводимые пользователем сохраняются в базе данных и наоборот информация которая хранится в базе данных извлекается для вывода пользователю. В SQL-запросах будем использовать основные функции такие как SELECT, INSERT INTO, UPDATE, DELETE, INNER JOIN (Рис. 19).

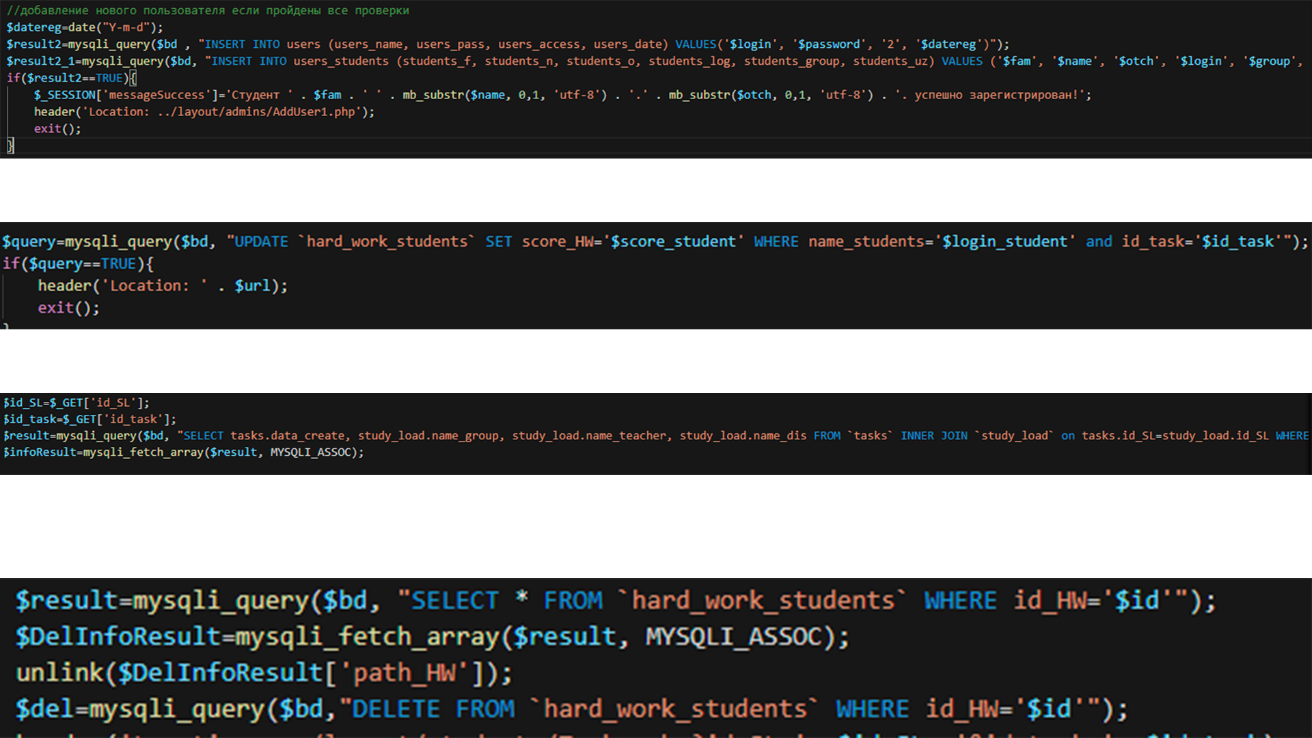


Рис. 19 Использование SQL-запросов

## Описание программных модулей

На данной платформе реализованы следующие программные модули:

* Модуль «Авторизация».

Модуль соединенный с базой данных, выполняющий проверку данных введенных пользователем для входа под аккаунт на платформу дистанционного обучения.

* Модуль «Добавить студента».

Модуль соединенный с базой данных, выполняющий запись новых пользователей с ролью «студент» в базу данных, встроенный в функционал администратора, позволяющий зарегистрированным пользователям произвести вход на платформу дистанционного обучения.

* Модуль «Добавить преподавателя».

Модуль соединенный с базой данных, выполняющий запись новых пользователей с ролью «преподаватель» в базу данных, встроенный в функционал администратора, позволяющий зарегистрированным пользователям произвести вход на платформу дистанционного обучения.

* Модуль «Добавить группу».

Модуль соединенный с базой данных, выполняющий добавление новой учебной группы в базу данных, встроенный в функционал администратора.

* Модуль «Добавить дисциплину».

Модуль соединенный с базой данных, выполняющий добавление новой дисциплины в базу данных, встроенный в функционал администратора.

* Модуль «Создать учебную нагрузку».

Модуль соединенный с базой данных, выполняющий добавление новой учебной нагрузке в базу данных, встроенный в функционал администратора.

* Модуль «Список учебной нагрузки».

Модуль соединенный с базой данных, выполняющий отображение структурированной информации из базы данных по всей добавленной учебной нагрузке, встроенный в функционал администратора.

* Модуль «Список пользователей».

Модуль соединенный с базой данных, выполняющий отображение структурированной информации из базы данных по всем зарегистрированным пользователям, встроенный в функционал администратора.

* Модуль «Создать задание».

Модуль соединенный с базой данных, выполняющий добавление нового задания по дисциплине для определенной учебной группы, с возможность загрузить методический материал, встроенный в функционал преподавателя.

* Модуль «Поставить оценку».

Модуль соединенный с базой данных, выполняющий добавление оценки к выполненной и загруженной работе студента, встроенный в функционал преподавателя.

* Модуль «Показать статистику».

Модуль соединенный с базой данных, выполняющий отображение статистики по проведенным занятиям, которые засчитываются только в том случае, когда преподаватель создал задание, встроенный в функционал преподавателя.

* Модуль «Создать отчет».

Модуль соединенный с базой данных, выполняющий автозаполнение excel таблицы информацией из базы данных, после заполнения таблицы, файл автоматически скачивается на компьютер преподавателя, встроенный в функционал преподавателя.

* Модуль «Добавить файл».

Модуль соединенный с базой данных, выполняющий добавление файла и загрузку этого файла на сервер, встроенный в функционал студента.

Рассмотрим каждый модуль подробнее.

### Модуль «Авторизация»

При переходе на платформу дистанционного обучения пользователя встречает форма авторизации, для входа на платформу необходимо ввести свои логин и пароль (Рис. 20).

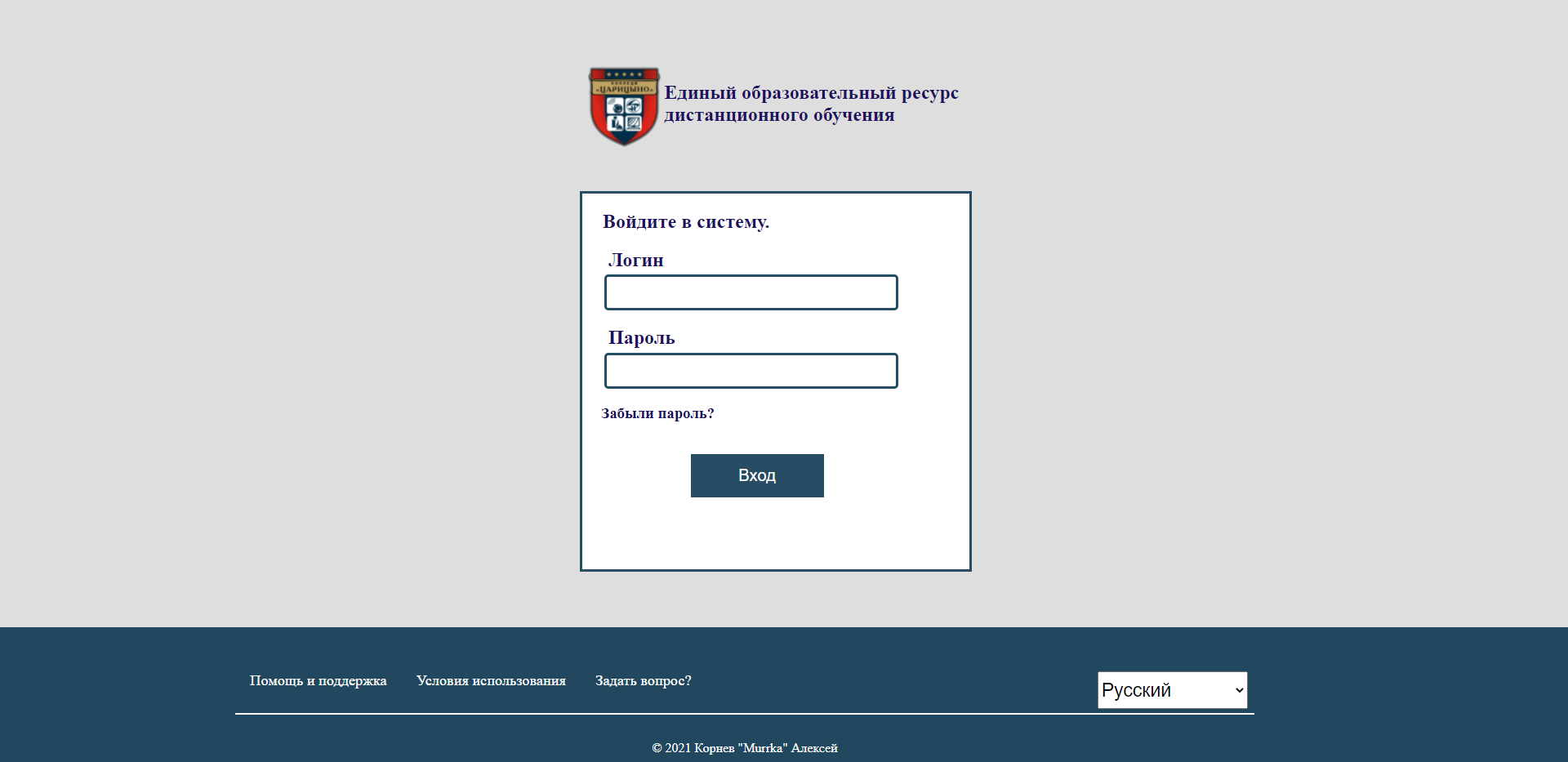
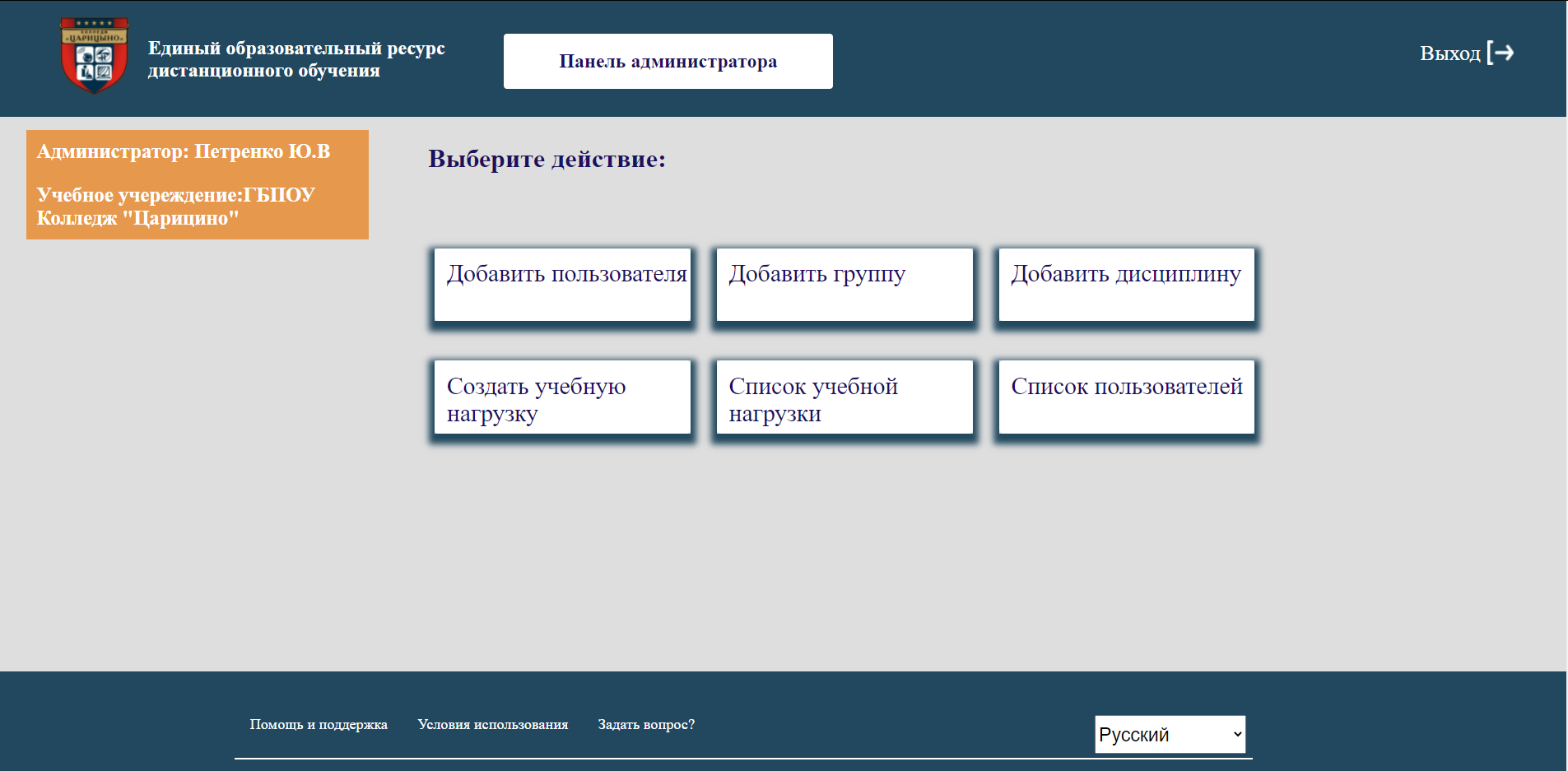


Рис. 20 Страница авторизации

1. Поле ввода логина пользователя, выданный администратором платформы дистанционного обучения.
2. Поле ввода пароля пользователя, выданный администратором платформы дистанционного обучения.
3. Кнопка «Вход», после ввода логина и пароля пользователь должен нажать на кнопку. После нажатия на кнопку пользователь будет перенаправлен на главную страницу его роли на платформе, в противном случае будет выведена ошибка понятная для пользователя.

На данной странице необходимо заполнить поля ввода логина и пароля, которые были выданы пользователю администратором. В противном случае выскочит ошибка с сообщением «Логин или пароль введены неверно». После успешной авторизации пользователь будет перенаправлен на главную страницу его роли в базе данных, на главную страницу администратора (Рис. 21), преподавателя (Рис. 22) или студента (Рис. 23). Код модуля представлен в приложении 3.

Рис. 21 Главная страница администратора

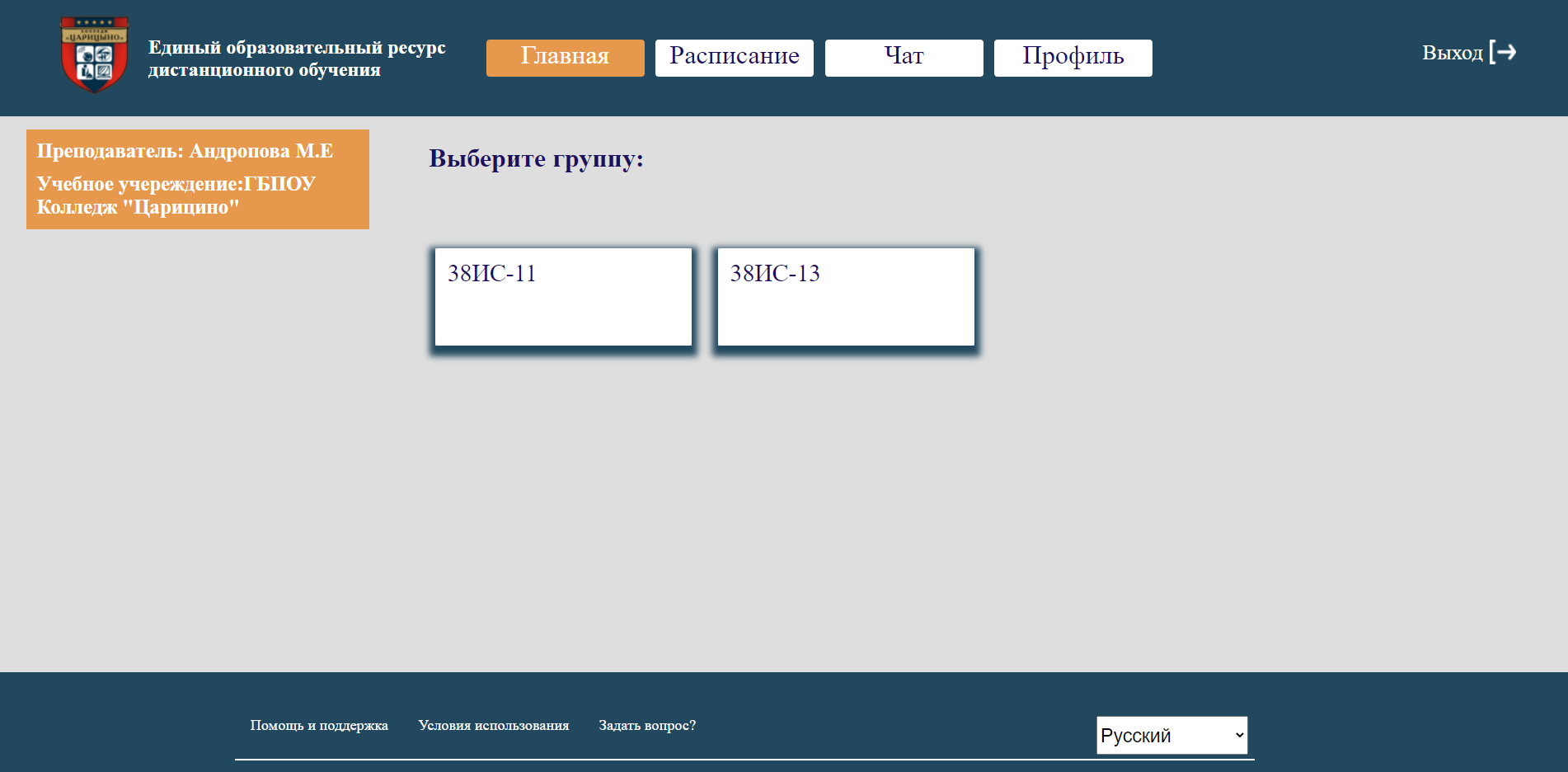


Рис. 22 Главная страница Преподавателя



Рис. 23 Главная страница студента

### Модуль «Добавить студента»

При переходе с главной страницы на страницу «Добавить пользователя» или при клике на левой функциональной панели на страницу «Добавить студента» администратор попадает на страницу с формой регистрации нового студента (Рис. 24).

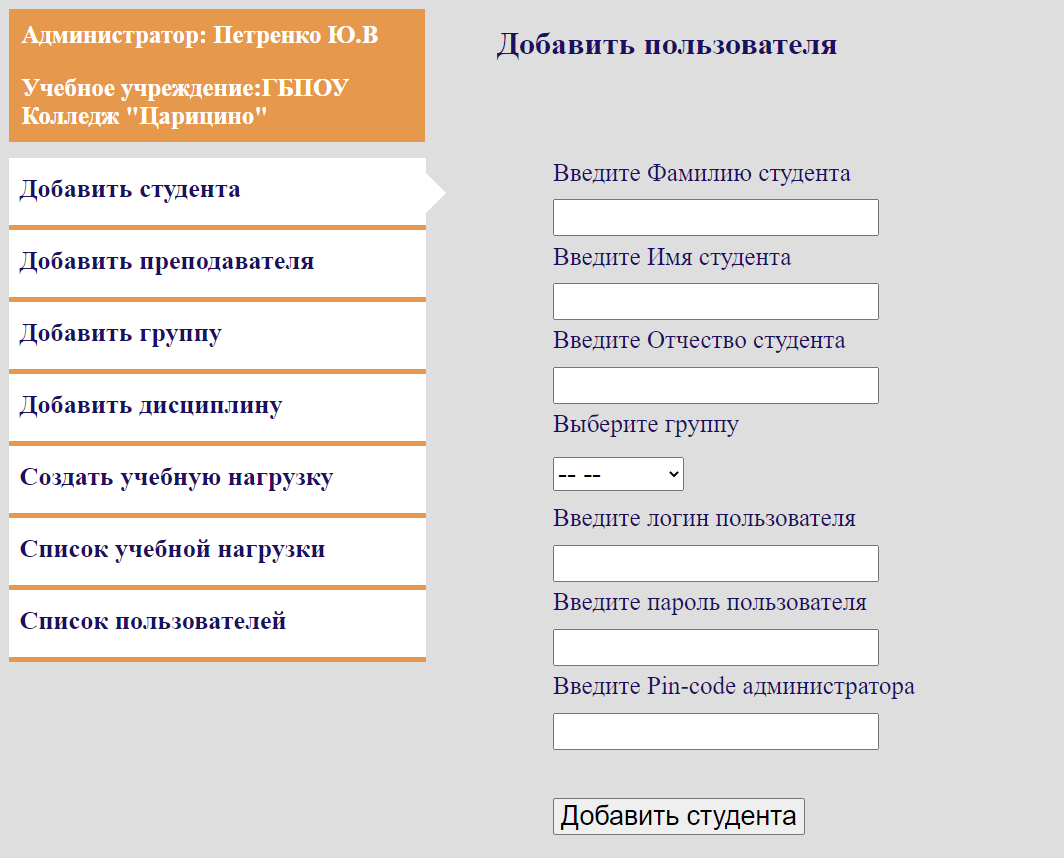


Рис. 24 Страница добавления нового студента

Поле ввода фамилии студента, обязательное поле для заполнения.

Поле ввода имени студента, обязательное поле для заполнения.

Поле ввода отчества студента, не обязательное поле для заполнения.

Список выбора уже добавленных групп, администратором, обязательное поле для заполнения. Если группа отсутствует, необходимо перейти на страницу «Добавить группу» и заполнить форму добавления там.

Поле ввода логина студента, обязательное поле для заполнения.

Поле ввода пароля студента, обязательное поле для заполнения.

Поле ввода пин-кода администратора, обязательное поле для заполнения. Пин-код администратора требуется для двухфакторной защиты, чтобы случайные данные не были добавлены в базу данных или при несанкционированном доступе к аккаунту администратора.

Кнопка «Добавить студента», при заполнении всех обязательных полей и нажатии на кнопку, администратору будет выведено окошко с успешным добавлением студента, в противном случае будет выведено окошко с ошибкой понятной для пользователя.

В данной форме администратору необходимо заполнить информацию о студенте, придумать ему логин и пароль. При неправильном заполнение формы для администратора выскочит окошко с ошибкой, сообщение будет зависеть от того какую ошибку он допустил (Рис. 25). При успешном заполнении формы для администратора выскочит окошко с оповещением о успешном добавление пользователя (Рис. 26). Код модуля представлен в приложении 3.

C:\Users\lexak\Desktop\Макеты\документация\для диплома\ошибка1.PNG

Рис.25 Ошибка заполнения формы добавления студента

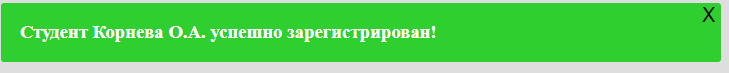


Рис. 26 Успешное добавление студента

### Модуль «Добавить преподавателя»

При клике на левой функциональной панели администратора на страницу «добавить преподавателя» администратор попадает на страницу с формой регистрации нового преподавателя (Рис. 27).

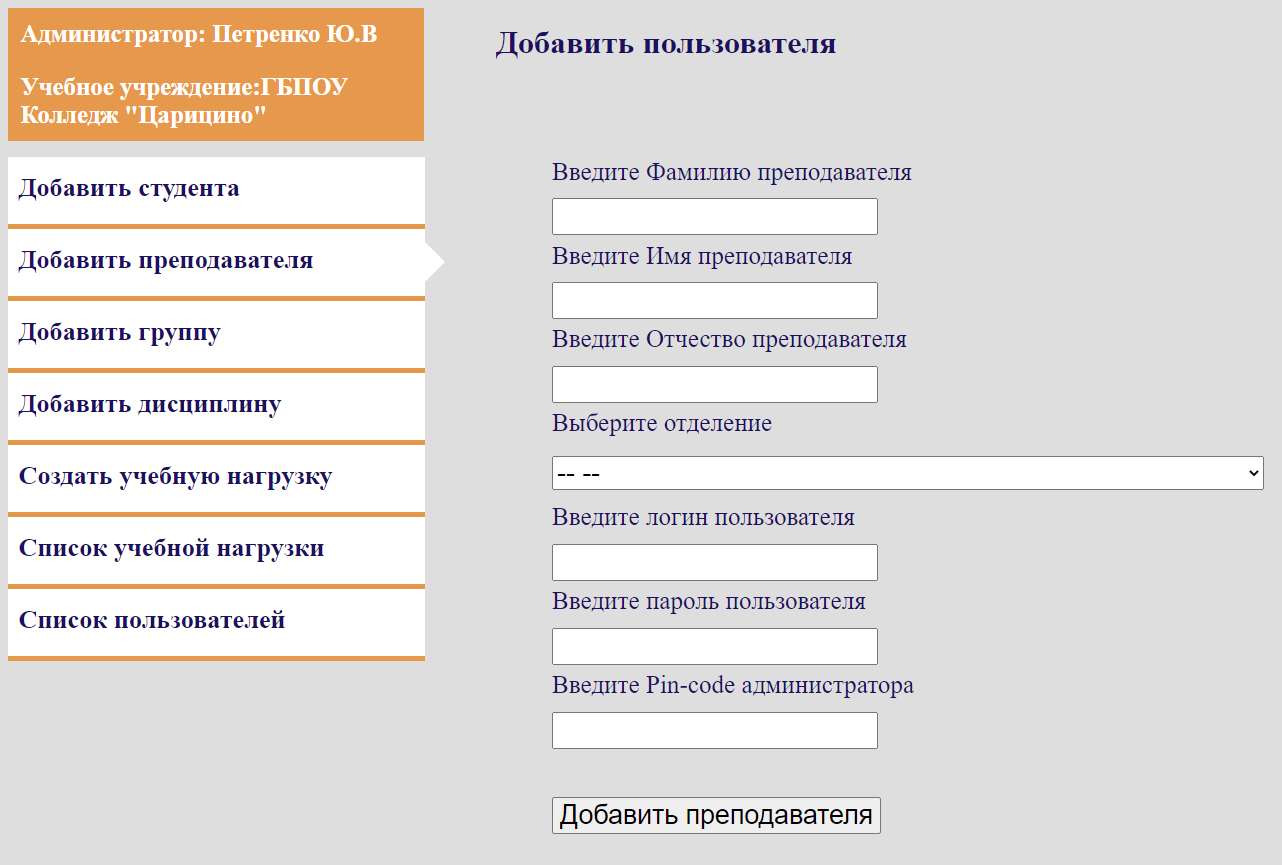


Рис. 27 Страница добавления нового преподавателя

Поле ввода фамилии преподавателя, обязательное поле для заполнения.

Поле ввода имени преподавателя, обязательное поле для заполнения.

Поле ввода отчества преподавателя, не обязательное поле для заполнения.

Список выбора отделения, на котором ведет занятия преподаватель, обязательное поле для заполнения.

Поле ввода логина преподавателя, обязательное поле для заполнения.

Поле ввода пароля преподавателя, обязательное поле для заполнения.

Поле ввода пин-кода администратора, обязательное поле для заполнения. Пин-код администратора требуется для двухфакторной защиты, чтобы случайные данные не были добавлены в базу данных или при несанкционированном доступе к аккаунту администратора.

Кнопка «Добавить преподавателя», при заполнении всех обязательных полей и нажатии на кнопку, администратору будет выведено окошко с успешным добавлением преподавателя, в противном случае будет выведено окошко с ошибкой понятной для пользователя.

В данной форме администратору необходимо заполнить информацию о преподавателе, придумать ему логин и пароль. При неправильном заполнение формы для администратора выскочит окошко с ошибкой, сообщение будет зависеть от того какую ошибку он допустил (Рис. 28). При успешном заполнении формы для администратора выскочит окошко с оповещением о успешном добавление пользователя (Рис. 29). Код модуля представлен в приложении 3.



Рис. 28 Ошибка заполнения формы добавления преподавателя

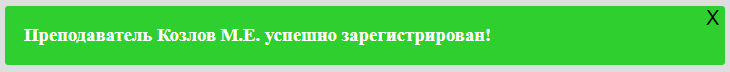


Рис. 29 Успешное добавление преподавателя

### Модуль «Добавить группу»

При переходе с главной страницы на страницу «Добавить группу» или при клике на левой функциональной панели на страницу «Добавить группу» администратор попадает на страницу с формой добавления новой группы в базу данных (Рис. 30).

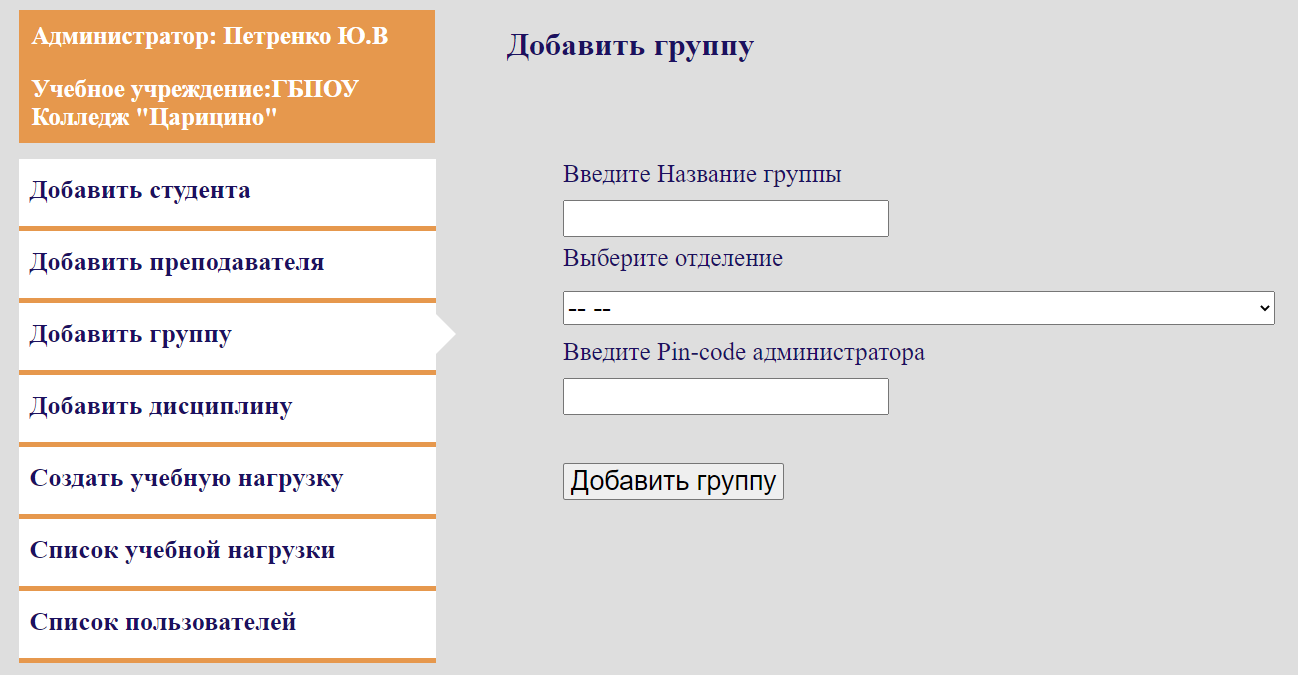


Рис. 30 Страница добавления учебной группы

Поле ввода названия учебной группы, обязательное поле для заполнения.

Список выбора отделения, на котором учиться группа, обязательное поле для заполнения.

Поле ввода пин-кода администратора, обязательное поле для заполнения. Пин-код администратора требуется для двухфакторной защиты, чтобы случайные данные не были добавлены в базу данных или при несанкционированном доступе к аккаунту администратора.

Кнопка «Добавить группу», при заполнении всех обязательных полей и нажатии на кнопку, администратору будет выведено окошко с успешным добавлением учебной группы, в противном случае будет выведено окошко с ошибкой понятной для пользователя.

В данной форме администратору необходимо заполнить информацию о учебной группе, ввести название и отделение, на котором группа обучается. При неправильном заполнение формы для администратора выскочит окошко с ошибкой, сообщение будет зависеть от того какую ошибку он допустил (Рис. 31). При успешном заполнении формы для администратора выскочит окошко с оповещением о успешном добавление группы (Рис. 32). Код модуля представлен в приложении 3.



Рис. 31 Ошибка заполнения формы добавления группы

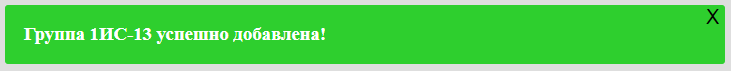


Рис. 32 Успешное добавление учебной группы

### Модуль «Добавить дисциплину»

При переходе с главной страницы на страницу «Добавить дисциплину» или при клике на левой функциональной панели на страницу «Добавить дисциплину» администратор попадает на страницу с формой добавления новой дисциплины в базу данных (Рис. 33).

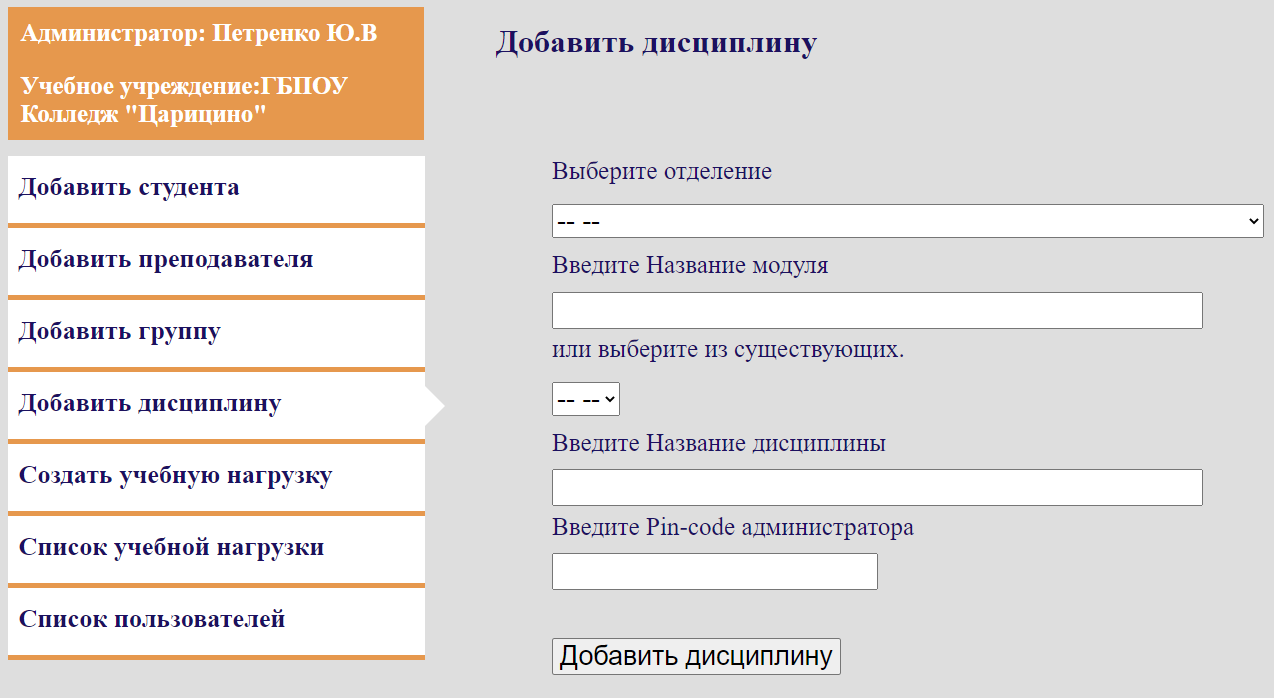


Рис. 33 Страница добавления дисциплины

Список выбора отделения, на котором ведется дисциплина, обязательное поле для заполнения.

Поле ввода названия модуля, к которому принадлежит дисциплина, обязательное поле для заполнения в случае если в выпадающем списке не выбрана уже добавленный модуль.

Список выбора модуля, если модуль уже был создан, его можно выбрать в выпадающем списке, обязательное поле для заполнения в случае если поле ввода название модуля не заполнено.

Поле ввода названия дисциплины, обязательное поле для заполнения.

Поле ввода пин-кода администратора, обязательное поле для заполнения. Пин-код администратора требуется для двухфакторной защиты, чтобы случайные данные не были добавлены в базу данных или при несанкционированном доступе к аккаунту администратора.

Кнопка «Добавить дисциплину», при заполнении всех обязательных полей и нажатии на кнопку, администратору будет выведено окошко с успешным добавлением дисциплины, в противном случае будет выведено окошко с ошибкой понятной для пользователя.

В данной форме администратору необходимо заполнить информацию о дисциплине, выбрать отделение, на котором дисциплина будет изучаться, ввести название модуля или выбрать из уже добавленных, ввести название дисциплины. При неправильном заполнение формы для администратора выскочит окошко с ошибкой, сообщение будет зависеть от того какую ошибку он допустил (Рис. 34). При успешном заполнении формы для администратора выскочит окошко с оповещением о успешном добавление группы (Рис. 35). Код модуля представлен в приложении 3.

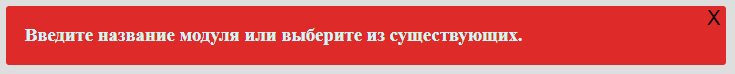


Рис. 34 Ошибка заполнения формы добавления дисциплины

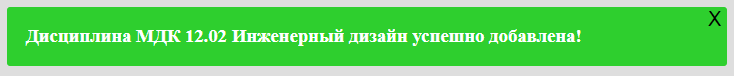


Рис. 35 Успешное добавление дисциплины

### Модуль «Создать учебную нагрузку»

При переходе с главной страницы на страницу «Создать учебную нагрузку» или при клике на левой функциональной панели на страницу «Создать учебную нагрузку» администратор попадает на страницу с формой добавления новой учебной нагрузки в базу данных (Рис. 36).

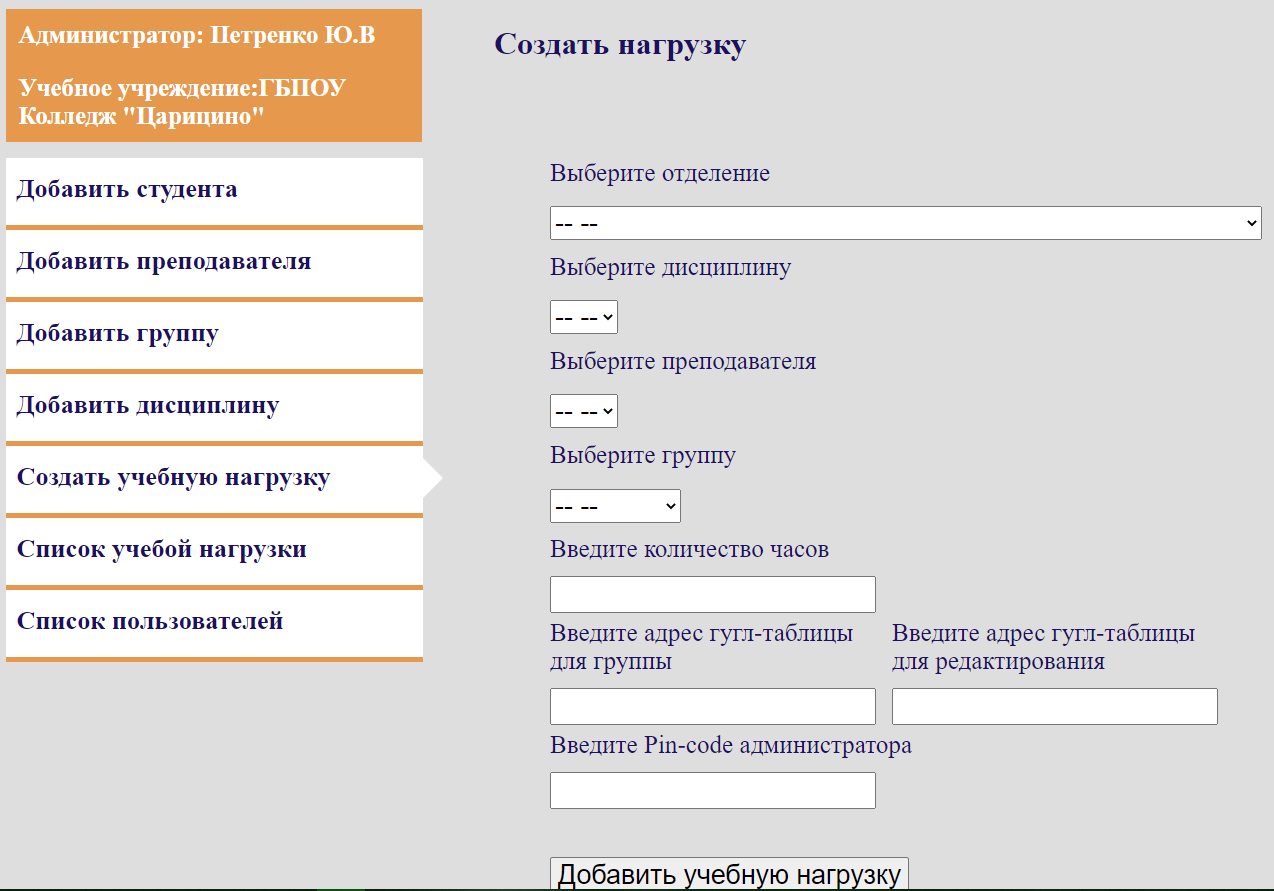


Рис. 36 Страница создания учебной нагрузки

Список выбора отделения, на котором будут проводиться занятия, обязательное поле для заполнения.

Список выбора дисциплины, по которой будут проводиться занятия, обязательное поле для заполнения.

Список выбора преподавателя, который будет проводить занятия, обязательное поле для заполнения.

Список выбора группы, в которой будут проводиться занятия, обязательное поле для заполнения.

Поле ввода количества часов по учебному плану, для выбранной дисциплины и закрепленного преподавателя, обязательное поле для заполнения.

Поле ввода url ссылки на гугл таблицу с оценками по дисциплине для внедрения ее на платформу и отображения для выбранной группы, обязательное поле для заполнения.

Поле ввода url ссылки на редактирование гугл таблицы с оценками по дисциплине для внедрения ее на платформу и возможности редактирования ее по ссылке, обязательное поле для заполнения.

Поле ввода пин-кода администратора, обязательное поле для заполнения. Пин-код администратора требуется для двухфакторной защиты, чтобы случайные данные не были добавлены в базу данных или при несанкционированном доступе к аккаунту администратора.

Кнопка «Добавить учебную нагрузку», при заполнении всех обязательных полей и нажатии на кнопку, администратору будет выведено окошко с успешным добавлением учебной нагрузки, в противном случае будет выведено окошко с ошибкой понятной для пользователя.

В данной форме администратору необходимо заполнить информацию об учебной нагрузке, выбрать отделение, на котором дисциплина будет проводиться, выбрать дисциплину, которая будет проводиться, выбрать преподавателя, который будет проводить дисциплину, выбрать группу, которая будет изучать дисциплину, ввести количество часов по учебному плану для выбранной дисциплины на семестр, ввести ссылку на гугл таблицу для внедрения ее в личные кабинеты студентов, которые принадлежат выбранной группе, ввести ссылку на редактирование гугл таблицы чтобы предоставить доступ выбранному преподавателю заполнять таблицу с оценками по выбранной дисциплине. При неправильном заполнение формы для администратора выскочит окошко с ошибкой, сообщение будет зависеть от того какую ошибку он допустил (Рис. 37). При успешном заполнении формы для администратора выскочит окошко с оповещением о успешном добавление учебной нагрузки (Рис. 38). Код модуля представлен в приложении 3.

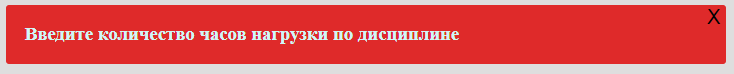


Рис. 37 Ошибка заполнения формы добавления учебной нагрузки

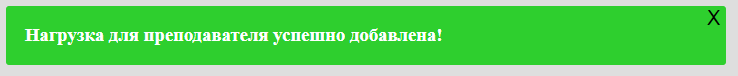


Рис. 38 Успешное добавление учебной нагрузки

### Модуль «Список учебной нагрузки»

При переходе с главной страницы на страницу «Список учебной нагрузки» или при клике на левой функциональной панели на страницу «Список учебной нагрузки» администратор попадает на страницу с таблицей добавленных ранее учебных нагрузок, выгруженных из базы данных (Рис. 39).



Рис. 39 Страница создания учебной нагрузки

На данной странице представлена вся учебная нагрузка, добавленная администратором, основная информация, это название дисциплины, преподаватель, который ведет дисциплину, группа которая изучается дисциплину и возможность удалить дисциплину при необходимости. Код отображения данных на странице представлен в приложении 1 или см. Рис. 40

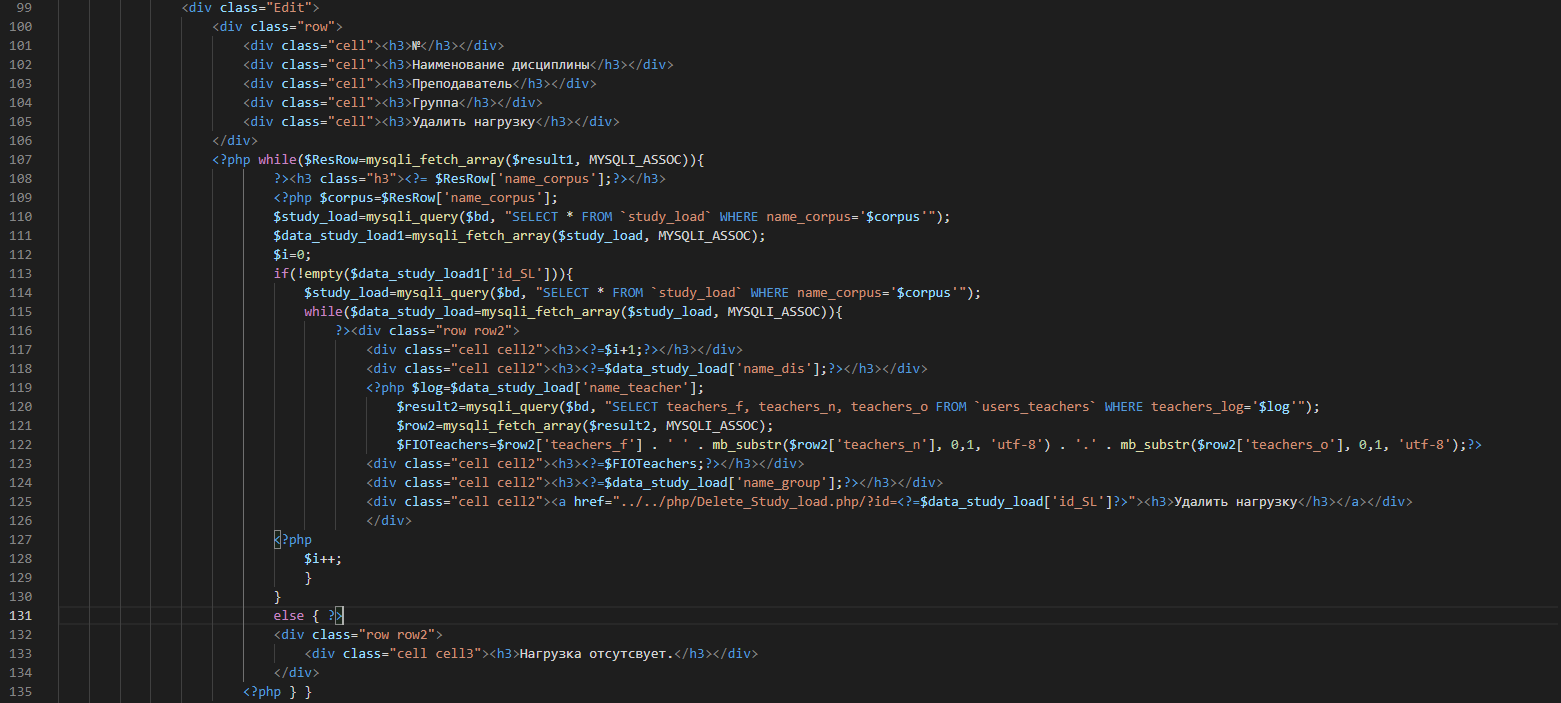


Рис. 40 Кусок кода заполнения информации из базы данных на странице «Список учебной нагрузки»

### Модуль «Список пользователей»

При переходе с главной страницы на страницу «Список пользователей» или при клике на левой функциональной панели на страницу «Список пользователей» администратор попадает на страницу с таблицей добавленных ранее пользователей, выгруженных из базы данных (Рис. 41 и 42).

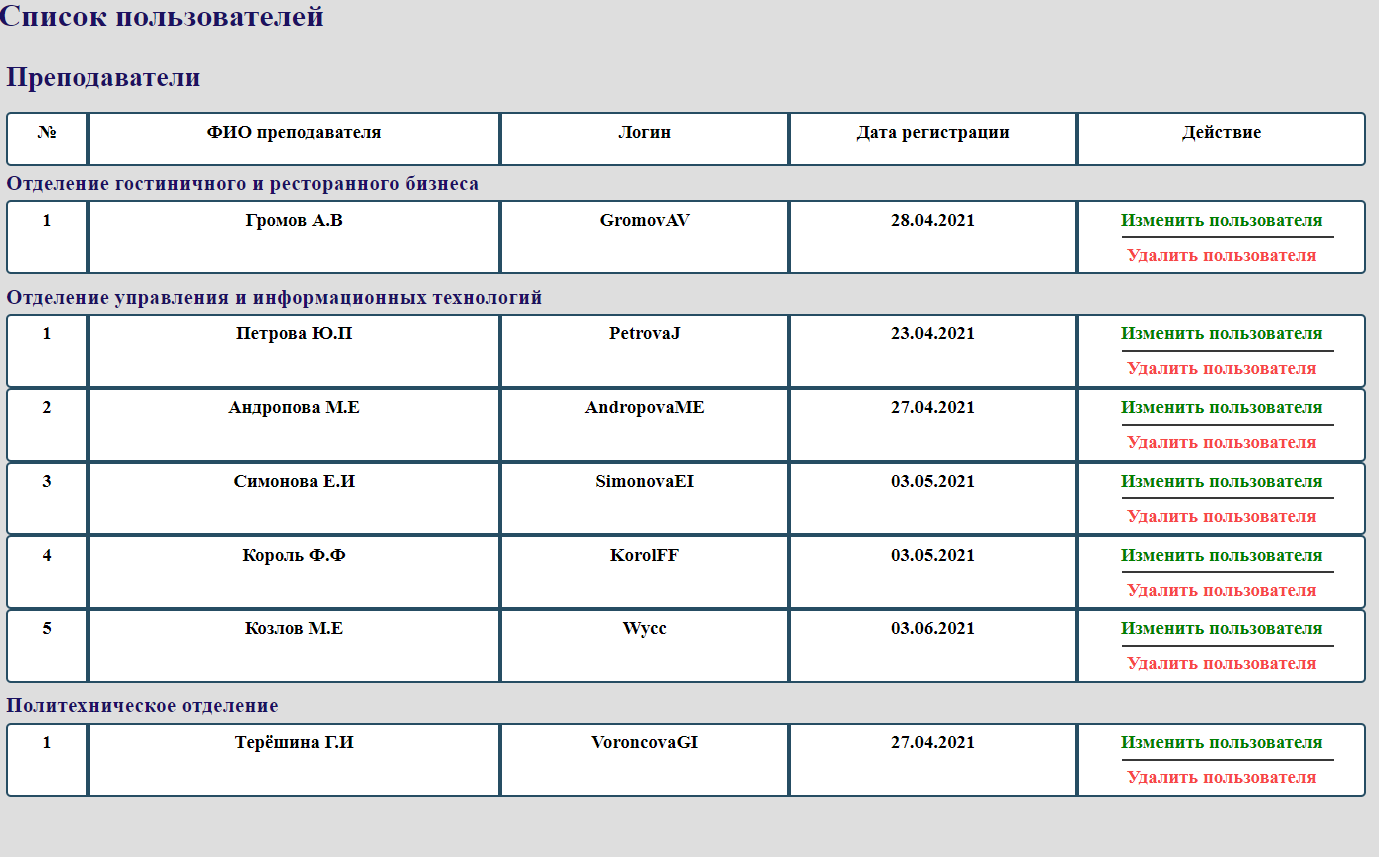


Рис. 41 Страница списка пользователей. Таблица преподаватели

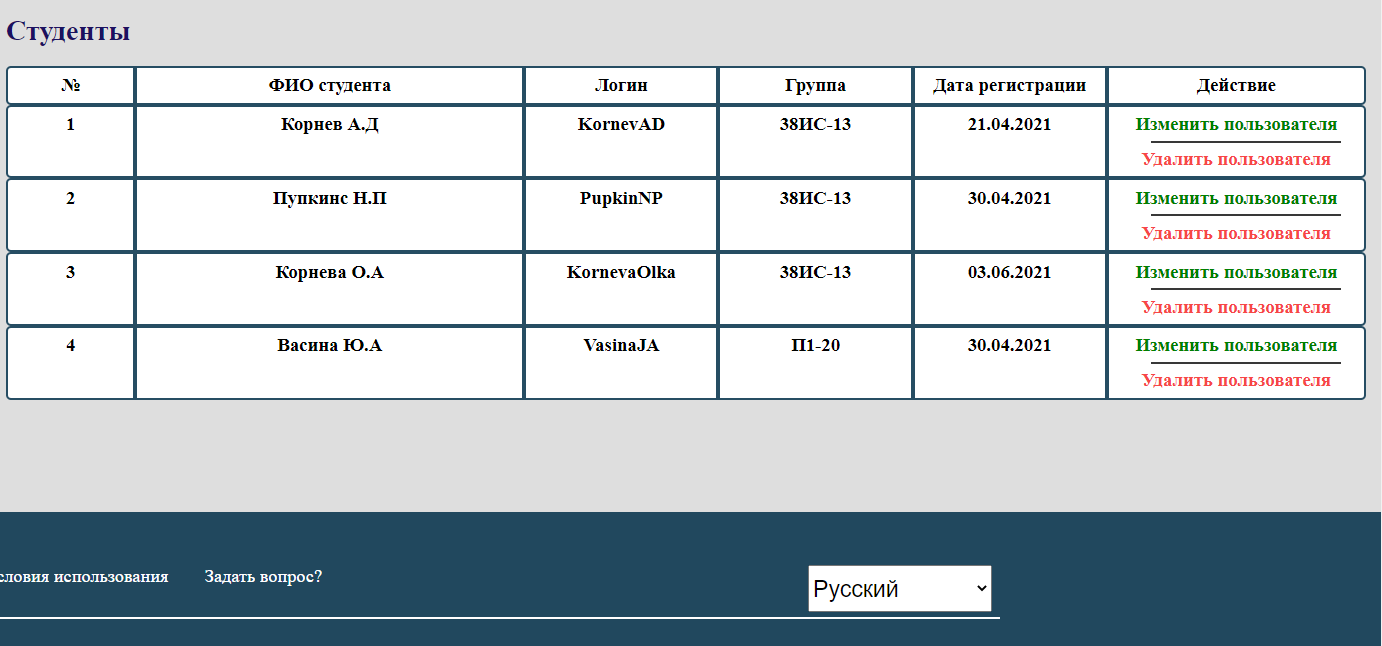


Рис. 42 Страница списка пользователей. Таблица студенты

На данной странице представлены все зарегистрированные пользователи администраторами учебного заведения. В таблице преподаватели представлена основная информация, ФИО преподавателя, логин, дата регистрации и возможность редактирования данных пользователя в случае смены фамилии, имени, утери пароля, переводе на другое отделение. А также возможность удаление учетной записи преподавателя. Код отображения данных на странице представлен в приложении 1.

При нажатии на ссылку «Изменить пользователя» администратор попадает на страницу изменения профиля пользователя (Рис. 43 и 44).

Поле ввода фамилии преподавателя или студента, поле автоматически заполнено данными выбранного пользователя.

Поле ввода имени преподавателя или студента, поле автоматически заполнено данными выбранного пользователя.

Поле ввода отчества преподавателя или студента, поле автоматически заполнено данными выбранного пользователя.

Список выбора отделения или группы, на котором ведет занятия преподаватель, поле автоматически заполнено данными выбранного пользователя.

Поле с отображением логина преподавателя или студента, поле автоматически заполнено данными выбранного пользователя изменять его нельзя.

Поле ввода пароля пользователя, обязательное поле для заполнения в случае изменения пароля пользователя.

Поле ввода пин-кода администратора, обязательное поле для заполнения. Пин-код администратора требуется для двухфакторной защиты, чтобы случайные данные не были добавлены в базу данных или при несанкционированном доступе к аккаунту администратора.

Кнопка «Изменить профиль преподавателя», при изменении данных администратором, администратору будет выведено окошко с успешным добавлением преподавателя, в противном случае будет выведено окошко с ошибкой понятной для пользователя.

В данной форме у администратора есть возможность изменить при необходимости информацию о пользователе. Код модуля представлен в приложении 3.

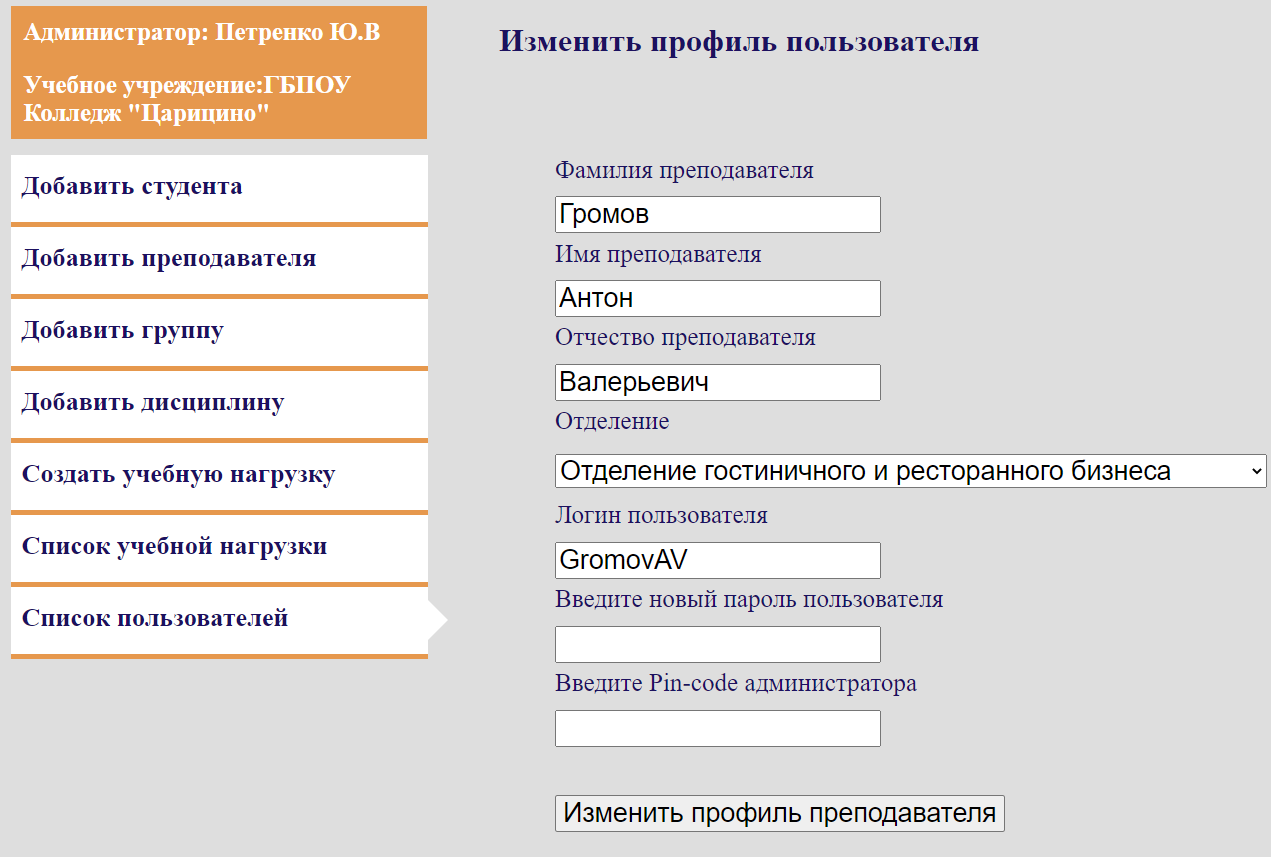


Рис. 43 Страница изменения профиля преподавателя

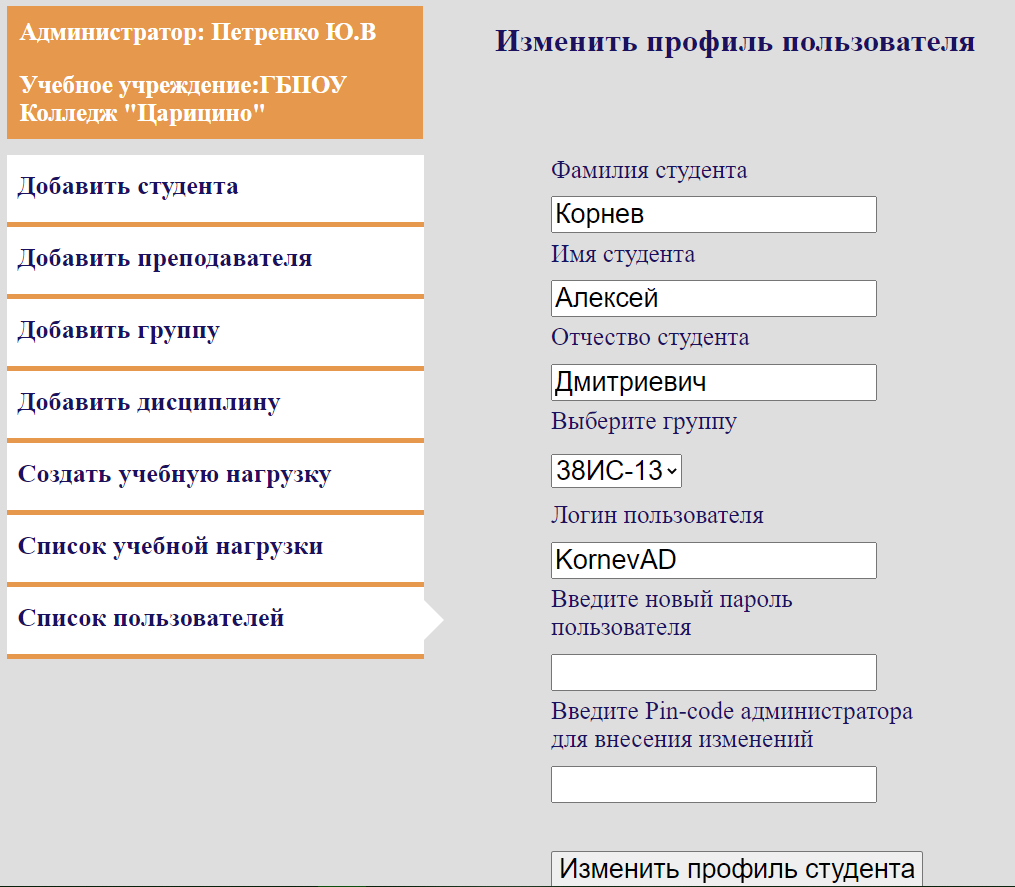


Рис. 44 Страница изменения профиля студента

### Модуль «Создать задание»

После авторизации пользователь с правами преподавателя будет перенаправлен на главную страницу где ему предложено выбрать группу, в которой преподаватель введет дисциплины (Рис. 45).

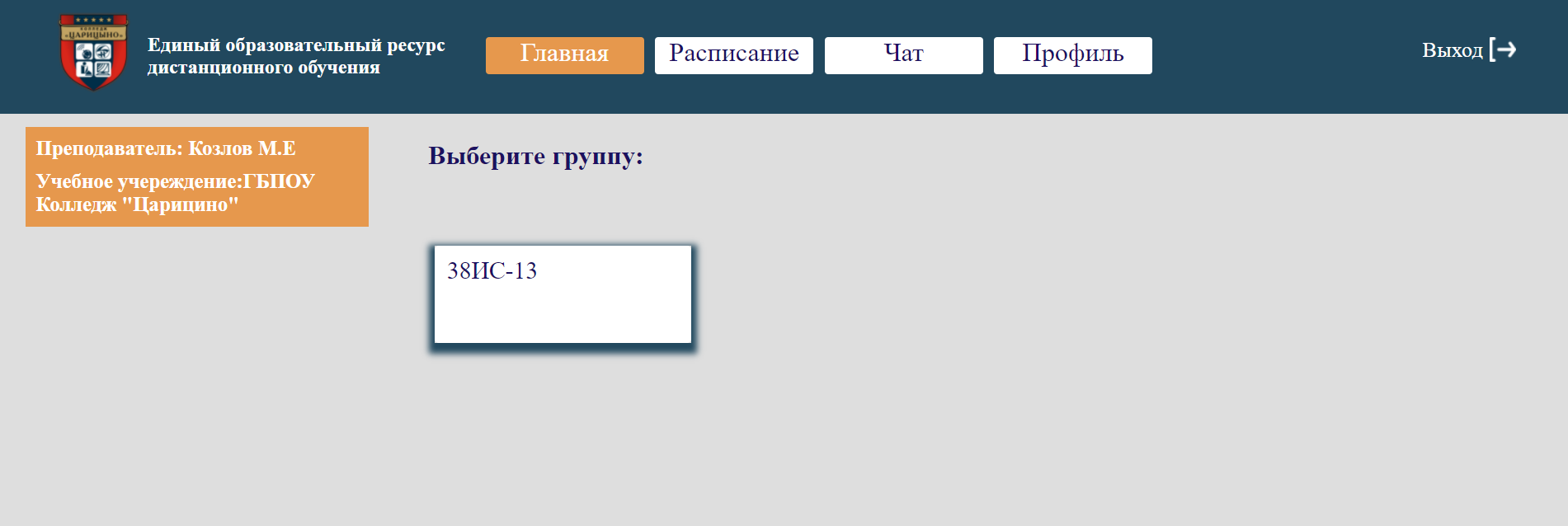


Рис. 45 Страница выбора группы

После выбора группы преподаватель будет перенаправлен на страницу выбор дисциплины, которую он ведет в выбранной группе (Рис. 46).

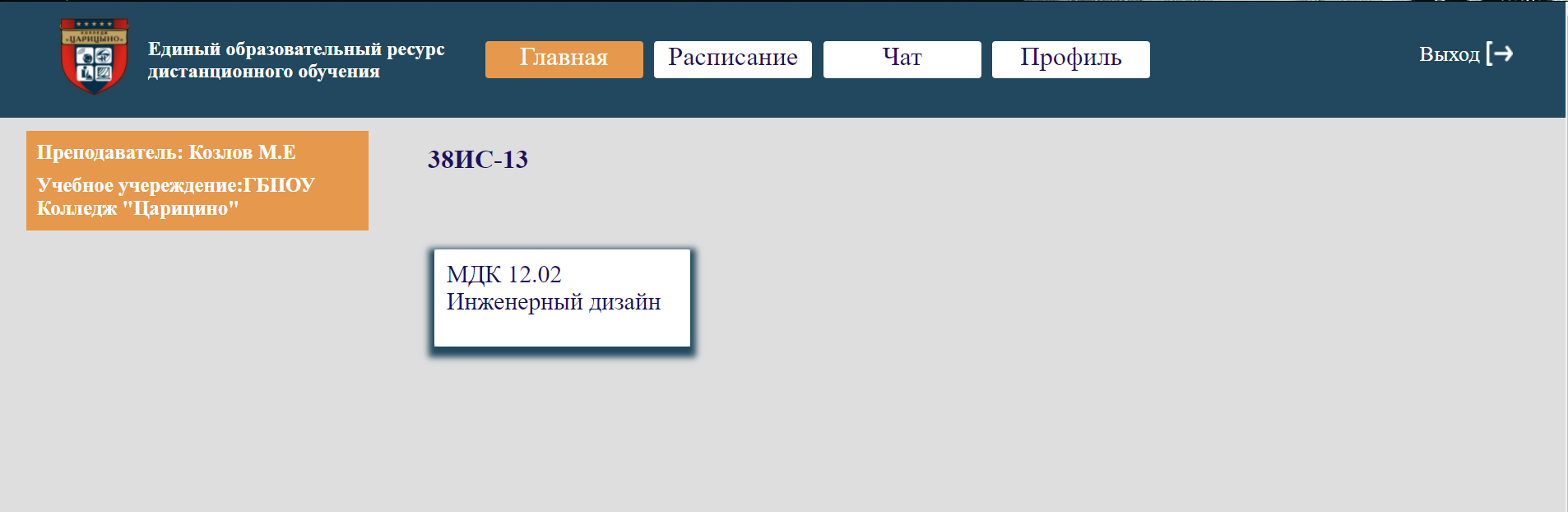


Рис. 46 Страница выбора дисциплины

После выбора дисциплины преподаватель попадает на главную страницу выбранной дисциплины, на которой отображена таблица с оценками и кнопка «Редактировать» (Рис. 47). При нажатии на кнопку «Редактировать» преподаватель будет перенаправлен на страницу гугл-таблицы с возможностью ее редактировать.

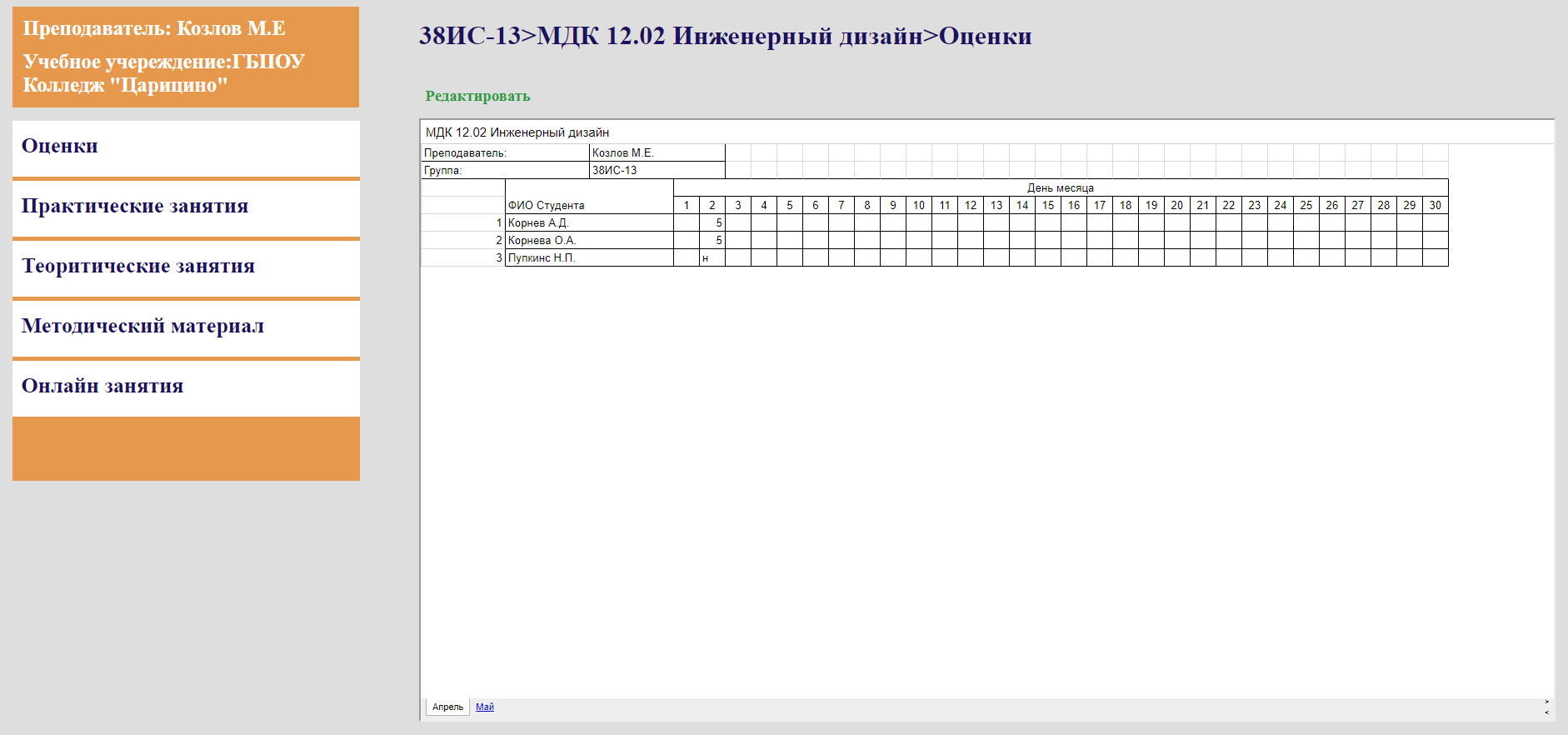


Рис. 47 Страница «Оценки» по выбранной дисциплине

При клике на левой функциональной панели на страницу «Практические занятия» или «Теоретические занятия» преподаватель попадает на страницу где отображаются все созданные практические или теоретические задания по выбранной дисциплине, а также присутствует кнопка «Создать задание» (Рис. 48).