Министерство образования и науки Республики Казахстан

Северо-Казахстанский университет им. М. Козыбаева

СЕМЕСТРОВЫЙ ПРОЕКТ 3

На тему: «Система для проведения «письменного» экзамена в СКУ»

ИКТ1АПО.01 СП

|  |  |
| --- | --- |
| Выполнили студенты  группы АПО-19 | Долгушин Н.Л.  Жантурин Д.Р.  Серикова Д.Л. |
| Научный руководитель  доцент, PhD | Астапенко Н.В. |

Петропавловск, 2022

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_Toc101948194)

[1 Аналитическая часть 5](#_Toc101948195)

[1.1 Исследование предметной области 5](#_Toc101948196)

[2 Проектирование системы для проведения экзамена 6](#_Toc101948197)

[2.1 Распределение задач 6](#_Toc101948198)

[2.2 Описание баз данных 10](#_Toc101948199)

[2.3 Разработка архитектуры программного средства 12](#_Toc101948200)

[2.4 Описание используемых алгоритмов 15](#_Toc101948201)

[2.5 Тестирование программного средства 20](#_Toc101948202)

[ЗАКЛЮЧЕНИЕ 28](#_Toc101948203)

[СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ 30](#_Toc101948204)

# ВВЕДЕНИЕ

На сегодняшний день в Северо-Казахстанском университете имени М. Козыбаева «письменные» экзамены проводятся в аудиториях и проверяются вручную. Такой метод проведения экзамена отнимает много времени у преподавателей и не отменяет того факта, что академическая честность может быть нарушена. Таким образом, стоит отметить, что предпосылками проведения экзамена онлайн являются: возможность ошибки при кодировании и декодировании логина студента, возможность нарушения академической честности, большие затраты по рабочему времени, трудовых и материальных ресурсов на ведение и контроль документов, поддержание данных в достоверном состоянии; неизбежно большое количество ошибок и описок при проведении выборки необходимых сведений и подготовке данных к различным отчетам.

Управление различными процессами при помощи компьютера позволяет добиться более высокой производительности труда и сэкономить массу времени. Высококачественная автоматизация технологических процессов значительно облегчает работу университета в целом. Преимущества, обычно приписываемые автоматизации, включают более высокие темпы производства и повышение производительности, более эффективное использование материалов, повышение качества продукции, повышение безопасности, сокращение рабочих недель и сокращение сроков производства. Увеличение объема производства и повышение производительности являются двумя основными причинами, оправдывающими использование автоматизации. Несмотря на заявления о высоком качестве, полученные от людей, автоматизированные системы, как правило, выполняют производственный процесс с меньшей изменчивостью, чем рабочие, что приводит к большему контролю и согласованности качества продукции.

Не будет преувеличением сказать, что сегодня системы онлайн-экзамена произвели настоящую революцию в обучении. Именно они помогли усовершенствовать оценивание знаний студентов. Теперь преподавателям гораздо проще проводить экзамены, а также следить за результатами и прогрессом своих учеников. Давно прошли дни, когда учителю надо было создавать каждый тест вручную и записывать результаты в журнал, высчитывать средний балл. Сегодняшние системы онлайн-экзамена помогают следить за прогрессом каждого ученика, избегая сложных подсчетов. Сами же инструменты для создания онлайн-экзаменов становятся всё понятнее и удобнее как для учеников, так и для преподавателя.

В статье «Организация онлайн-контроля средствами цифровых технологий», автором которой является Ивашкина Т.А., указано, что «… в виртуальном образовании возникла потребность в эффективном контроле самостоятельных действий студента во время проверки знаний. Важным является анализ онлайн-контроля средствами цифровых технологий, что позволяет преподавателям в режиме онлайн контролировать процесс тестирования студентов». В данном высказывании показывается актуальность темы семестрового проекта. Для решения данной проблемы и необходима разработка системы проведения «письменного» экзамена.

*Целью семестрового проекта* является создание системы для проведения «письменного» экзамена в СКУ.

Для разработки программного средства необходимо *решить задачи синтеза и анализа*. К задачам анализа относятся:

* анализ предметной области;
* выбор средств проектирования базы данных;
* выбор средств проектирования программного средства.

После решения задач анализа можно приступить к выполнению задач синтеза. К ним относятся:

* проектирование и создание базы данных;
* разработка архитектуры программного средства;
* реализация алгоритмов программного средства;
* отладка и тестирование программного средства.

*Практическая значимость данной работы* заключалась в использовании результатов работы в практической деятельности университета. Разработанная в ходе семестрового проекта система для проведения «письменного» экзамена в СКУ упростит задачу проведения экзаменов в университете.

# 1 Аналитическая часть

## 1.1 Исследование предметной области

Предметной областью семестровой работы является система для проведения «письменного» экзамена. Письменный экзамен может демонстрировать общие или специальные знания студента. Разрабатываемая система позволяет преподавателям размещать текст билетов по дисциплине и критерии оценивания. Заведующий просматривает и утверждает билеты. Студенту случайным образом достается билет по дисциплине. Студент пишет текст ответа или прикрепляет файл ответа. Преподаватель проверяет ответы на вопросы (не зная автора ответа) и выставляет оценки по установленным критериям. Для сдачи письменного экзамена каждый студент имеет личный логин, именуемый ИКС, и пароль для входа в автоматизированную информационную систему. В роли письменного экзамена может выступать эссе, билеты.

В создаваемой системе для проведения экзамена присутствует три класса пользователей: преподаватель, заведующий, студент. Класс «преподаватель» обладает следующими свойствами: размещение текста билетов, критериев билетов. Класс «заведующий» обладает свойствами: просмотр билетов, утверждение билетов. Класс «студент» случайным образом получает билет и может написать текст билета либо прикрепить файл. Информация о билетах, критериях, преподавателях, заведующих, студентах хранится в соответствующих базах данных. Каждый класс имеет доступ к системе для проведения «письменного» экзамена по своему логину (ИКС в системе университета) и паролю. Все данные содержатся в соответствующей базе данных.

Университет, представленный в автоматизированной экзаменационной системе, осуществляет проведение письменных и устных экзаменов. В данной семестровой работе разрабатывается система для проведения «письменного» экзамена. Разрабатываемая автоматизированная экзаменационная система должна содержать в себе процесс размещения билетов и критериев билета, просмотр и утверждение билетов, написание ответа на билет или прикрепление файла.

Экзамены в письменной форме проводятся в рамках соблюдения принципов Лиги Академической честности в целях итогового контроля учебных достижений обучающихся, оценки уровня освоения содержания дисциплины после ее изучения в соответствии с учебной программой.

Таким образом, для работы автоматизированной информационной системы необходимо разработать следующие алгоритмы: вход пользователя, размещение билетов и критериев билета, просмотр и утверждение билетов, написание ответа на билет или прикрепление файла.

# 2 Проектирование системы для проведения экзамена

## 2.1 Распределение задач

Группа, участвующая в семестровом проекте на тему: «Система для проведение «письменного» экзамена», состоит из 3-х студентов. Техническое задание представляет собой создание макетов проекта, исследование предметной области, проектирование и подключение баз данных, создание форм, алгоритмов, архитектуры проекта. Распределение технического задания представлено в таблице 1.

Для управления ресурсами, которое включает в себя планирование ресурсов и составление графика работ, была выбрана методология Scrum. Scrum обладает достаточно привлекательными достоинствами. Scrum ориентирован на клиента, адаптивен. Scrum дает клиенту возможность делать изменения в требованиях в любой момент времени (но не гарантирует того, что эти изменения будут выполнены). Возможность изменения требований привлекательна для многих заказчиков ПО.

Джефф Сазерленд, автор книги «Scrum. Революционный метод управления проектами» выделил 8 шагов по использованию методики, которым следовала команда, а именно:

1. выбор Scrum-мастера;
2. определение цели проекта и ожидаемых результатов;
3. сбор команды для разработки;
4. описание требований к продукту;
5. планирование спринтов;
6. организация «meet-up» (ежедневные обсуждения о том, кто, что сделал вчера, что сделал сегодня и что мешает выполнить задачу);
7. обзоры рабочих частей продукта;
8. проведение ретроспективы (обсуждение проблемы и решение после каждого спринта).

Основой Scrum является Sprint, в течении которого выполняется работа над продуктом. По окончанию Sprint должна быть получена новая рабочая версия продукта. Sprint всегда ограничен по времени (1-4 недели) и имеет одинаковую продолжительность на протяжении всей жизни продукта. В семестровом проекте спринт занимает 1 неделю. Scrum подразумевает три роли: владелец продукта, Scrum-мастер и участники команды разработчиков. Владельцем продукта является команда проекта. В роли Scrum-мастера выступает капитан команды – Серикова Дарья.

Перед началом каждого Sprint производится Sprint Planning – планирование, на котором производится оценка содержимого Product Backlog (то, что уже разработано) и формирование Sprint Backlog, который содержит задачи, которые должны быть выполнены в текущем спринте. Каждый спринт должен иметь цель, которая является мотивирующим фактором и достигается с помощью выполнения задач. Все задачи расписаны по спринтам в таблице 1.

Таблица 1 Распределение обязанностей технического задания

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Неделя  Студент | 10 неделя | 11 неделя | 12 неделя | 13 неделя | 14 неделя |
| Серикова Дарья | Написание введения, распределение обязанностей, исследование предметной области | Описание методологии Scrum, описание архитектуры проекта | Описание алгоритмов проекта | Создание формы для выставления оценки | Описание списка использованной литературы |
| Долгушин Никон | Создание макетов проекта, реализация структуры MVC | Создание алгоритма авторизации | Создание формы для добавления ответа на билет | Создание формы для размещения билетов и критериев | Сбор архитектуры проекта |
| Жантурин Даниял | Загрузка проекта в GitHub, реализация структуры MVC, создание макетов проекта | Проектирование баз данных | Подключение БД проекта, создание формы входа | Создание формы для просмотра и утверждения билетов | Тестирование проектного средства |

Каждый день производится Daily Scrum, на котором каждый член команды отвечает на вопросы «что я сделал вчера?», «что я планирую сделать сегодня?», «какие препятствия на своей работе я встретил?». Задача Daily Scrum – определение статуса и прогресса работы над Sprint, раннее обнаружение возникших препятствий, выработка решений по изменению стратегии, необходимых для достижения целей Sprint'а.

По окончанию Sprint'а производятся Sprint Review и Sprint Retrospective, задача которых оценить эффективность (производительность) команды в прошедшем Sprint'е, спрогнозировать ожидаемую эффективность (производительность) в следующем спринте, выявлении имеющихся проблем, оценки вероятности завершения всех необходимых работ по продукту и другое. Scrum достаточно прост в изучении, позволяет экономить время, за счет исключения не критичных активностей. Scrum позволяет получить потенциально рабочий продукт в конце каждого Sprint'а.  
Scrum делает упор на самоорганизующуюся, многофункциональную команду, способную решить необходимые задачи с минимальной координацией. Это особенно привлекательно для малых компаний и стартапов, так как избавляет от необходимости от найма или обучения специализированного персонала руководителей.

Scrum является адаптивной методологией, требующей вдумчивого применения, но в качестве ее безусловных преимуществ можно назвать:

* Возможность быстрого запуска проекта с наиболее приоритетными функциями и минимально возможным бюджетом;
* Ежедневный контроль над ходом работ, и более гибкий контроль над бюджетом проекта;
* Частые демонстрации проекта. Применение данной методологии предполагает регулярную демонстрацию разработок заказчику (заказать эффективный сайт можно, скажем, тут, заодно сможете подсчитать стоимость в калькуляторе), что позволяет в будущем избежать полного провала работы команды и разочарований клиента;
* Возможность вносить коррективы в техническое задание по ходу реализации проекта, что является несомненным преимуществом для заказчика.

Отчет по окончанию каждого спринта описан в следующих разделах по мере выполнения заданий.

Составление отчета включает в себя написание введения, аналитической части, проектной части, заключения, а также добавления списка использованной литературы. Аналитическая часть включает в себя исследование предметной области, проектная часть – описание баз данных, архитектуры и алгоритмов проекта, тестирование программного средства. Распределение обязанностей по выполнению отчета представлено в таблице 2.

Таблица 2 Распределение обязанностей выполнения отчета

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Неделя  Студент | 10 неделя | 11 неделя | 12 неделя | 13 неделя | 14 неделя |
| Серикова Дарья | Оформление введения, распределения обязанностей, исследования предметной области | Описание методологии Scrum | Описание всех алгоритмов | Добавление формы ответов в раздел тестирования | Добавление в отчет заключения, списка использованной литературы |
| Долгушин Никон | Описание алгоритмов проекта | Описание алгоритма авторизации | Добавление формы входа в раздел тестирования | Добавление формы для размещения билетов в раздел тестирования | Форматирование отчета |
| Жантурин Даниял | Описание баз данных проекта | Описание архитектуры проекта | Добавление формы для утверждения билетов в раздел тестирования | Добавление формы выставления оценки | Описание тестирования проектного средства |

Первый спринт проекта включает в себя оформление введения, распределение обязанностей, исследование предметной области, описание баз данных проекта. После распределения работ по спринтам длительностью в неделю можно приступить к выполнению задания в следующем порядке: первым этапом является проектирование баз данных. Спроектированные базы данных в дальнейшем должны быть описаны в архитектуре программного средства. Следующим этапом является разработка общих алгоритмов проекта. После подключения баз данных и сборки алгоритмов и архитектуры проекта, проводится тестирование программного средства.

## 2.2 Описание баз данных

Первый спринт включает в себя описание баз данных проекта. Система для проведения экзамена, как и любая другая, имеет необходимость в хранении и обработке данных. Использование баз данных позволяет своевременно получать доступ к информации, хранящейся внутри БД, а также сохранять и изменять все данные по определенным запросам. Такой способ хранения информации гарантирует то, что данные не потеряются. При разработке системы использовалась база данных пользователей с классами пользователей: студент, преподаватель, заведующий. База данных пользователей содержит ИКС, пароль и соответствующий статус пользователя (студент, преподаватель, заведующий), по которому в системе для пользователя уставлены соответствующие права. При добавлении ответа на билет ИКС студента кодируется и не является видимым для преподавателя или заведующего в целях соблюдения академической честности.

Инфологическая модель БД – это модель, которая описывает данные предметной области с использованием естественного языка. Проектирование инфологической модели является основной задачей при создании БД. Цель инфологической модели- обеспечение наиболее естественных для человека способов сбора и представления той или иной информации, которую предполагается хранить в создаваемой базе. Поэтому инфологическую модель данных пытаются строить по аналогии с естественным языком (последний не может быть использован в чистом виде из-за сложности компьютерной обработки текстов и неоднозначности любого естественного языка). Основными конструктивными элементами инфологических моделей являются сущности, связи между ними и их свойства. Хранение информации происходит в структурированном виде в виде таблиц.

На инфологической модели представлены данные кафедр, факультетов, дисциплин, специальностей, содержащие id, полное и короткое наименование кафедры, факультета, дисциплины, специальности, наименование на казахском и английском языках. Базы данных студентов и преподавателей содержат id, код студента для соблюдения академической честности, ФИО, id группы.

Типы экзаменов заполняются в базу данных controltypes. Она содержит id, наименование типов экзаменов на русском, казахском и английском языках. Экзамены и экзамены для групп находятся в базах данных controls и controlsforgroups. В них находятся id экзамена, экзамена для групп, дисциплины, группы, преподавателя, дату экзамена, номер семестра. База данных tickets содержит добавленные преподавателем билеты на «письменный» экзамен, критерии также заполняются в базу данных criteria.

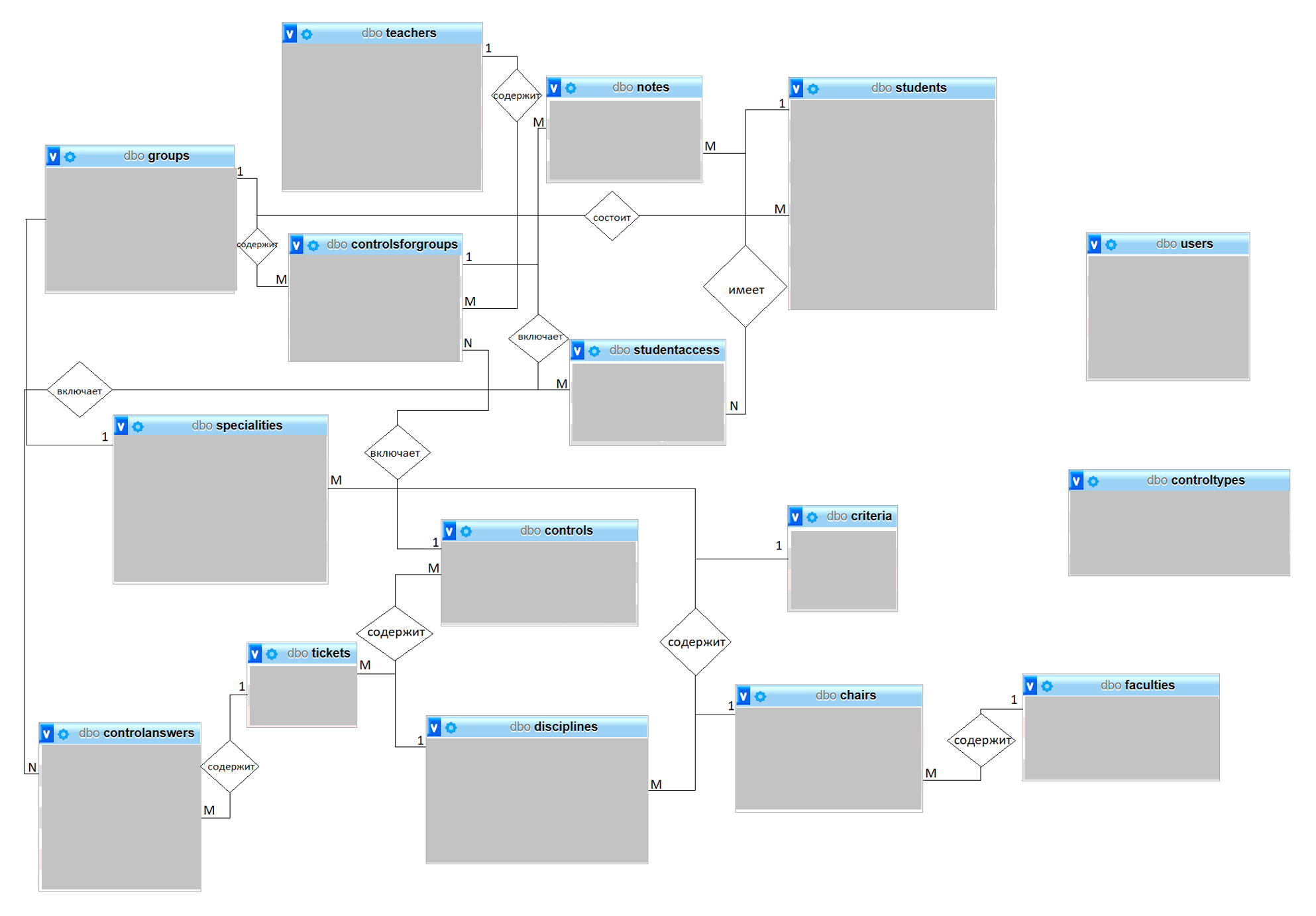


Рисунок 2.1 Инфологическая модель данных

После построения инфологической модели можно приступать к построению даталогической модели. Даталогическая (концептуальная) модель БД – это логическая организация данных и их взаимосвязь. Структурирование данных выполняется в соответствии с выбранной СУБД, которая строится на основе выбранной модели представления данных: иерархической, сетевой или реляционной. На данном этапе проектировщик создает структуру данных и организует связь между объектами. Проектирование даталогической модели – важный этап в проектировании БД. На этом этапе важно правильно выделить сущности и описать их атрибуты. Ошибка на этом этапе может обернуться разработчику значительными потерями времени и сил в дальнейшем.

Даталогическая модель баз данных представлена в соответствии с рисунком 2.2.

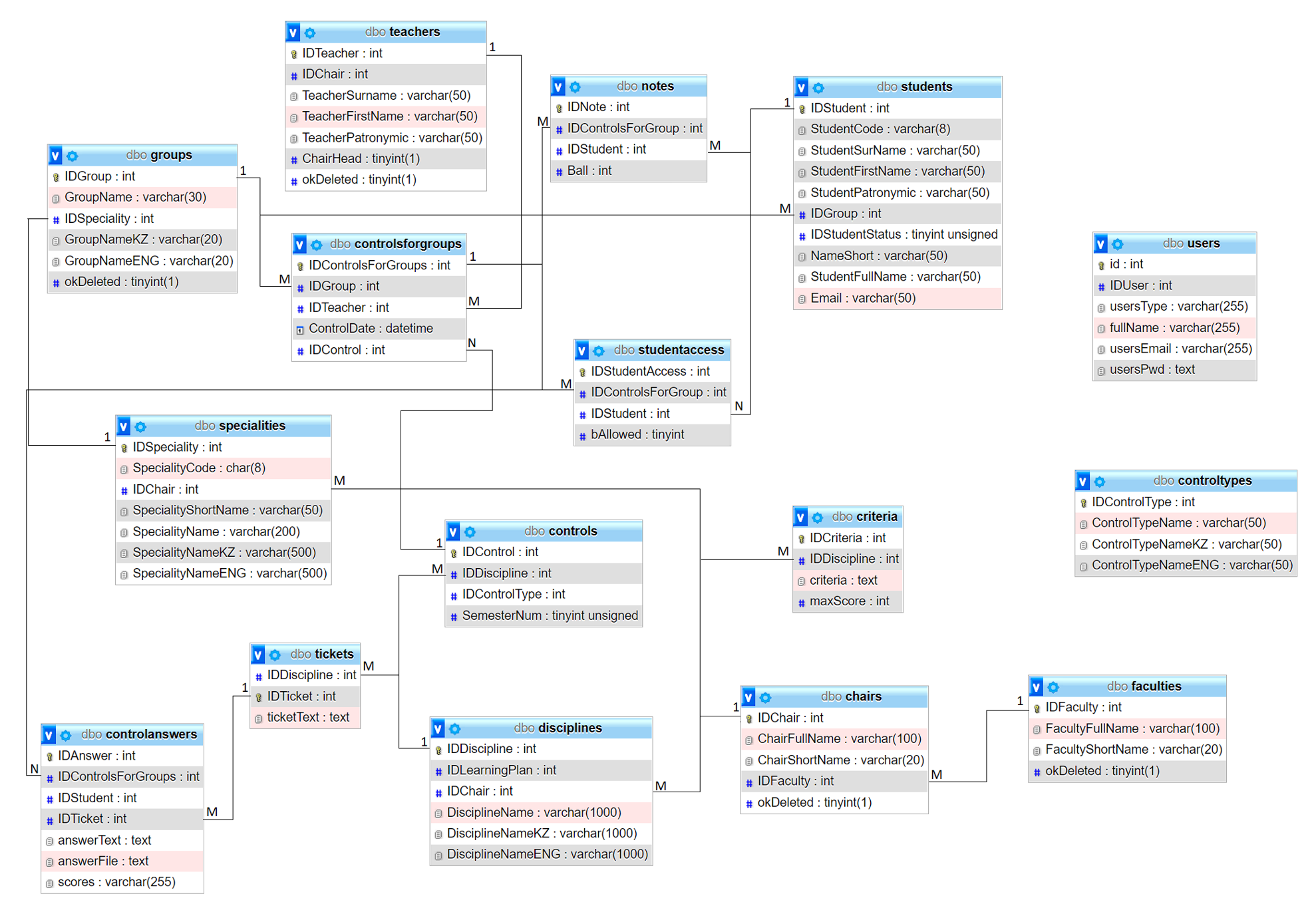


Рисунок 2.2 Даталогическая модель данных

В семестровом проекте в роли хранилища баз данных выступает MySQL, свободная реляционная система управления базами данных. Таким образом, в системе для проведения «письменного» экзамена участвуют базы данных пользователей, преподавателей и заведующих, студентов, дисциплин, групп, факультетов, кафедр, билетов, критериев, ответов на билеты, а также оценок.

Таким образом, первый спринт был успешно завершен. Результатами первого спринта являются инфологическая и даталогическая модели баз данных, оформленное введение, распределенные работы по спринтам, а также анализ предметной области.

## 2.3 Разработка архитектуры программного средства

Второй спринт включает в себя описание архитектуры программного средства. Архитектура программного обеспечения системы отображает организацию или структуру системы и дает объяснение того, как она ведет себя. Система представляет собой набор компонентов, выполняющих определенную функцию или набор функций. Другими словами, архитектура программного обеспечения обеспечивает прочную основу, на которой может быть построено программное обеспечение.

Разработка архитектуры является первым этапом борьбы со сложностью ПС, на котором реализуется принцип выделения относительно независимых компонентов. Архитектура проекта представляет собой распределенные по различным папкам согласно структуре Model-View-Controller (MVC, «Модель-Представление-Контроллер», «Модель-Вид-Контроллер»), которая представляет собой схему разделения данных приложения и управляющей логики на три отдельных компонента: модель, представление и контроллер — таким образом, что модификация каждого компонента может осуществляться независимо.

Для двустороннего взаимодействия между преподавателем и студентом необходимо разработать динамические веб-страницы. Для этого используется модель MVC, которая представляет собой способ организации кода, при котором предполагается выделение блоков, отвечающих за решение разных задач. Один блок отвечает за данные, другой блок – за внешний вид, а третий контролирует работу приложения.

Model – компонент, отвечающий за данные, а также определяющий структуру приложения. View – компонент, отвечающий за взаимодействие с пользователем, таким образом, он определяет вид приложения и способы его использования. Controller – компонент, отвечающий за связь между model и view. Код данного компонента определяет, как сайт реагирует на действия пользователя. В разработке системы для проведения «письменного» экзамена в СКУ моделью будет являться сайт или приложение, на котором проводится экзамен.

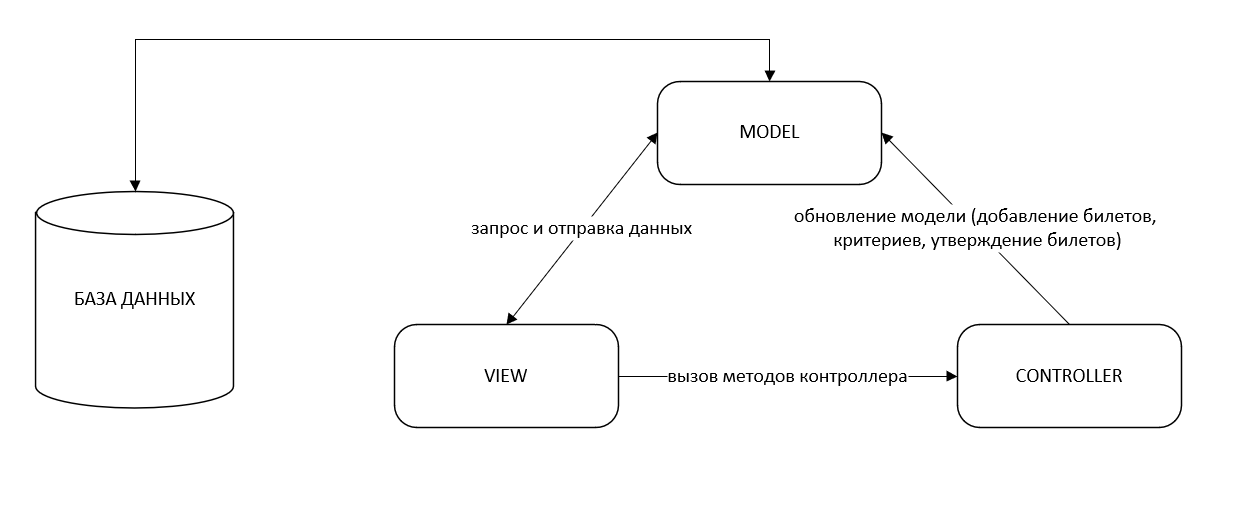


Рисунок 2.3 MVC Архитектура

На рисунке 2.3 представлена архитектура программного средства, на которой можно увидеть, что разрабатываемый проект содержит в себе три основных интерфейса: авторизация, страница со стороны студента и страница со стороны преподавателя и заведующего. Каждый интерфейс включает в себя несколько модулей. Страница студента состоит из модуля просмотра билета и добавления ответа на билет. Также студент может просмотреть свои оценки. Страница преподавателя включает в себя модуль размещения билетов, утверждения их и выставления оценки.

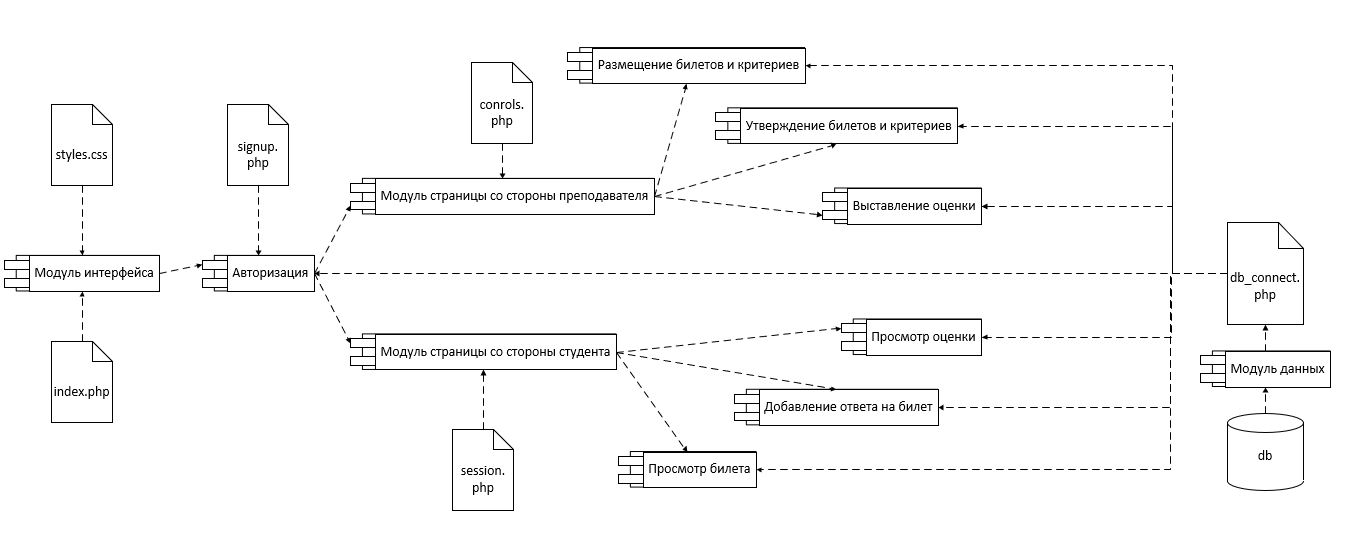


Рисунок 2.4 Архитектура проекта

Результатами второго спринта являются схема архитектуры проекта, описанная методология Scrum.

## 2.4 Описание используемых алгоритмов

Второй спринт также включает в себя описание алгоритма авторизации. В разработанной системе используются следующие алгоритмы: вход в систему, добавление билетов и критериев для преподавателя, просмотр и утверждение билетов и критериев для заведующего, просмотр билета и добавление ответа для студента. Алгоритм входа в систему происходит следующим образом:

* Начало алгоритма;
* Объявление параметров. Ввод данных, где iks соответствует ИКС пользователя в системе СКУ, password – пароль в системе, установленный заранее для всех;
* Проверка данных в соответствии с базой данных пользователей, если данные верны – вход в систему, если нет – вывод ошибки;
* Закрытие окна;
* Конец алгоритма.

Представление алгоритмов происходит посредством блок-схем. Блок-схема — это диаграмма, на которой обычно представлен процесс, система или компьютерный алгоритм и которая используется для документирования, планирования, уточнения или визуализации многоэтапного рабочего процесса. Благодаря блок-схемам вы можете определить цели и масштабы рабочего процесса, а также установить необходимые задачи в хронологическом порядке. В блок-схемах в том виде, в каком мы их знаем сегодня, используют символы различной формы для обозначения специфических элементов рабочих процессов, а стрелки и линии указывают на направление перехода от этапа к этапу.

Блок-схема алгоритма входа представлен на рисунке 2.5.

Таким образом, второй спринт был успешно завершен.

В программе помимо алгоритма входа есть алгоритм добавления билетов и критериев. Алгоритм происходит следующим образом:

* Начало алгоритма;
* Ввод данных о билетах и критериях;
* Импорт данных в БД «билеты и критерии»;
* Конец алгоритма.

Представление алгоритмов происходит, как и с алгоритмом регистрации посредством блок-схем. Блок-схема алгоритма добавления билетов и критериев представлена в соответствии с рисунком 2.6.

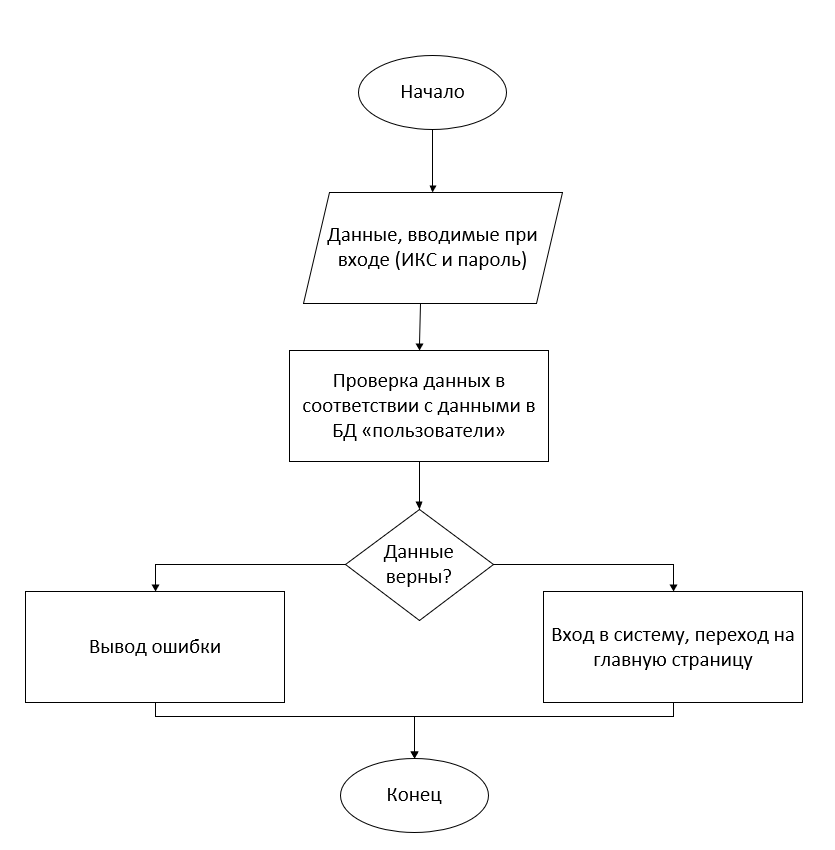


Рисунок 2.5 Блок-схема авторизации

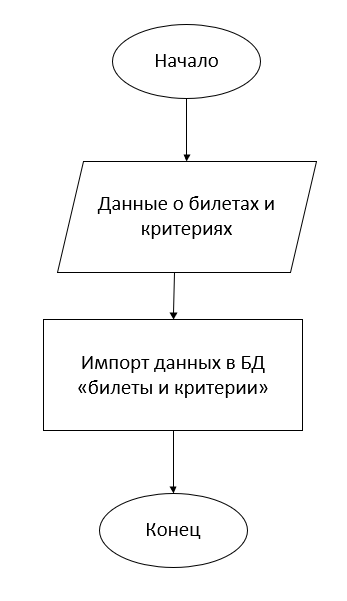


Рисунок 2.6 Алгоритм добавления билетов и критериев

После добавления билетов и критериев преподавателем их должен утвердить заведующий кафедры. Алгоритм происходит следующим образом:

* Начало алгоритма;
* Просмотр билетов и критериев;
* Утверждение билетов и критериев посредством нажатия кнопки «Утвердить»;
* Изменение статуса билетов и критериев БД на «Утверждено»;
* Конец алгоритма.

Представление алгоритмов происходит, как и с алгоритмом входа посредством блок-схем. Блок-схема алгоритма добавления билетов и критериев представлена в соответствии с рисунком 2.7.

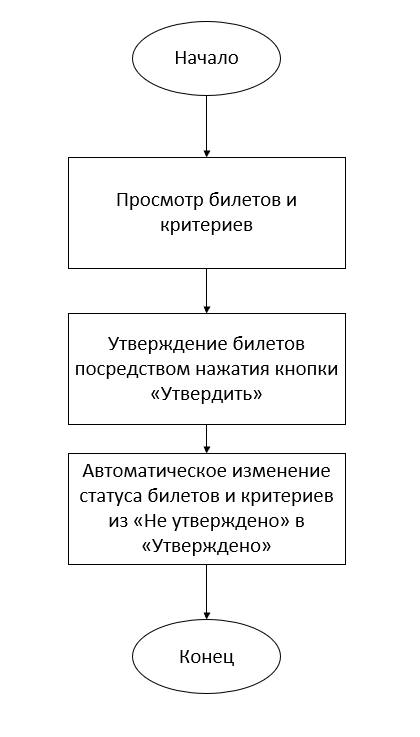


Рисунок 2.7 Алгоритм утверждения билетов

Следующим алгоритмом является добавление ответа на билет студентом. Алгоритм происходит следующим образом:

* Начало алгоритма;
* Просмотр случайным образом выданного билета и критериев;
* Добавление ответа на билет посредством написания ответа в поле, либо прикрепления файла;
* Добавление данных в БД;
* Конец алгоритма.

Представление алгоритмов происходит, как и с алгоритмом входа посредством блок-схем. Блок-схема алгоритма добавления ответа представлена в соответствии с рисунком 2.8.

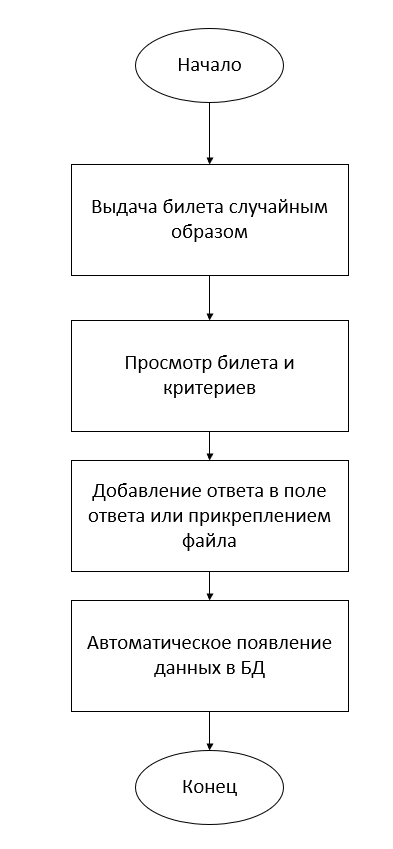


Рисунок 2.8 Добавление ответа на билет

Последним алгоритмом является выставление оценки преподавателем. Алгоритм происходит следующим образом:

* Начало алгоритма;
* Появление данных из БД на страницу для преподавателя;
* Просмотр ответов студента;
* Добавление оценки на билет;
* Конец алгоритма.

Представление алгоритмов происходит, как и с алгоритмом входа посредством блок-схем. Блок-схема алгоритма добавления оценки представлена в соответствии с рисунком 2.9.

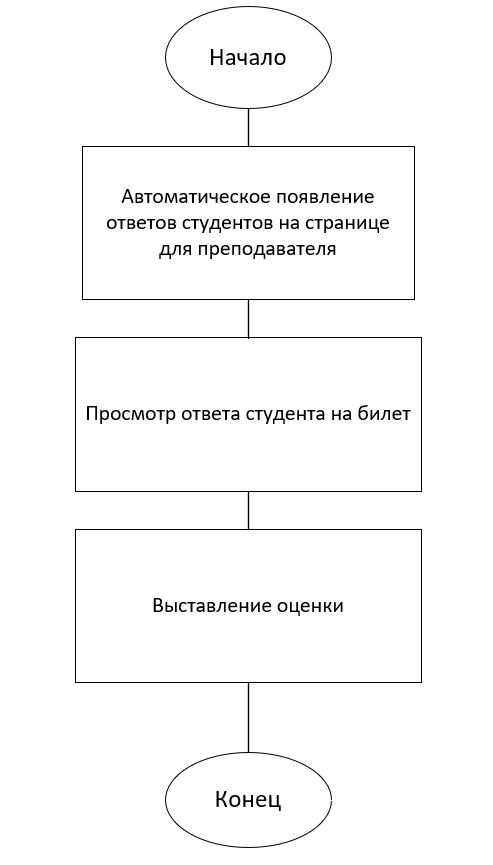


Рисунок 2.9 Выставление оценки

После подключения баз данных и создания необходимых алгоритмов можно приступить к сбору архитектуры проекта. Сбор архитектуры проекта включает в себя соединение всех страниц приложения. Последним этапом проектной разработки является тестирование программного средства.

Результатами третьего спринта являются описанные выше алгоритмы добавления билетов и критериев, утверждения билетов и критериев, добавления ответа на билет, выставления оценки.

## 2.5 Тестирование программного средства

Четвертый спринт включает в себя добавление формы ответов в раздел тестирования, формы для размещения билетов в раздел тестирования, формы выставления оценки, представленные ниже.

После сбора архитектуры проекта можно приступить к тестированию программного средства. При тестировании системы для проведения «письменного» экзамена можно убедиться в работоспособности кнопок для перехода между страницами, входа в систему, добавлении билетов и критериев, их утверждении, а также в добавлении ответа на билет. При запуске системы открывается главная страница, которая представлена на рисунке 2.10.

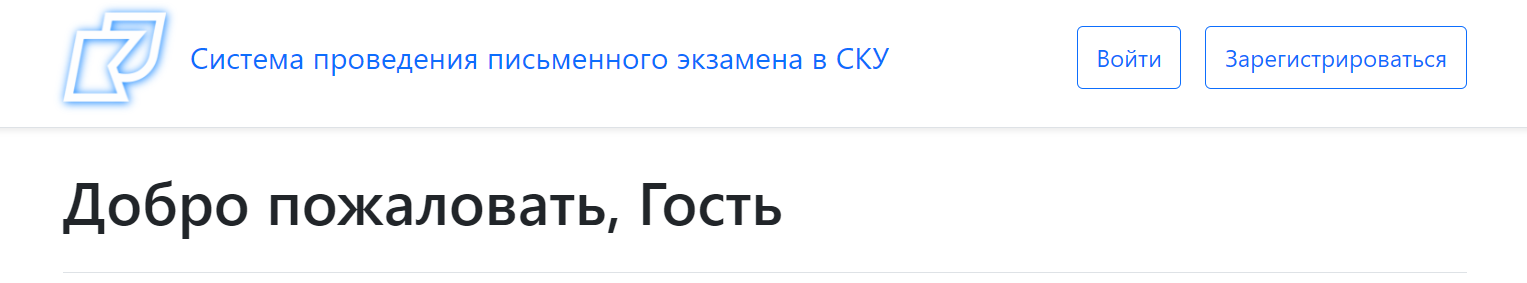


Рисунок 2.10 Главная страница

В верхней части страницы располагается заголовок, на котором с левой стороны расположен логотип сайта, который является ссылкой на главную страницу. С правой стороны расположены 2 ссылки, которые предоставляют пользователю возможность перейти на страницу входа и на страницу регистрации. Заголовок главной страницы представлен в соответствии с рисунком 2.11.

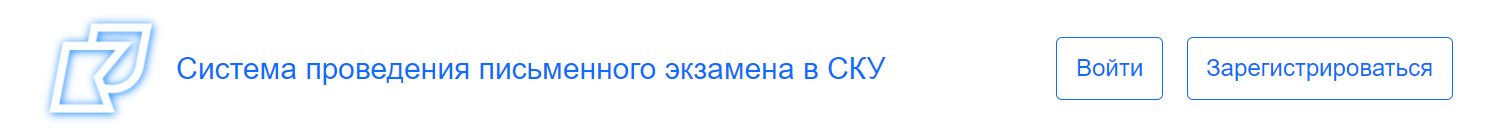


Рисунок 2.11 Заголовок страницы

При нажатии кнопки «Войти» открывается новое окно входа с полями электронной почты и пароля. Авторизация представлена на рисунке 2.12.

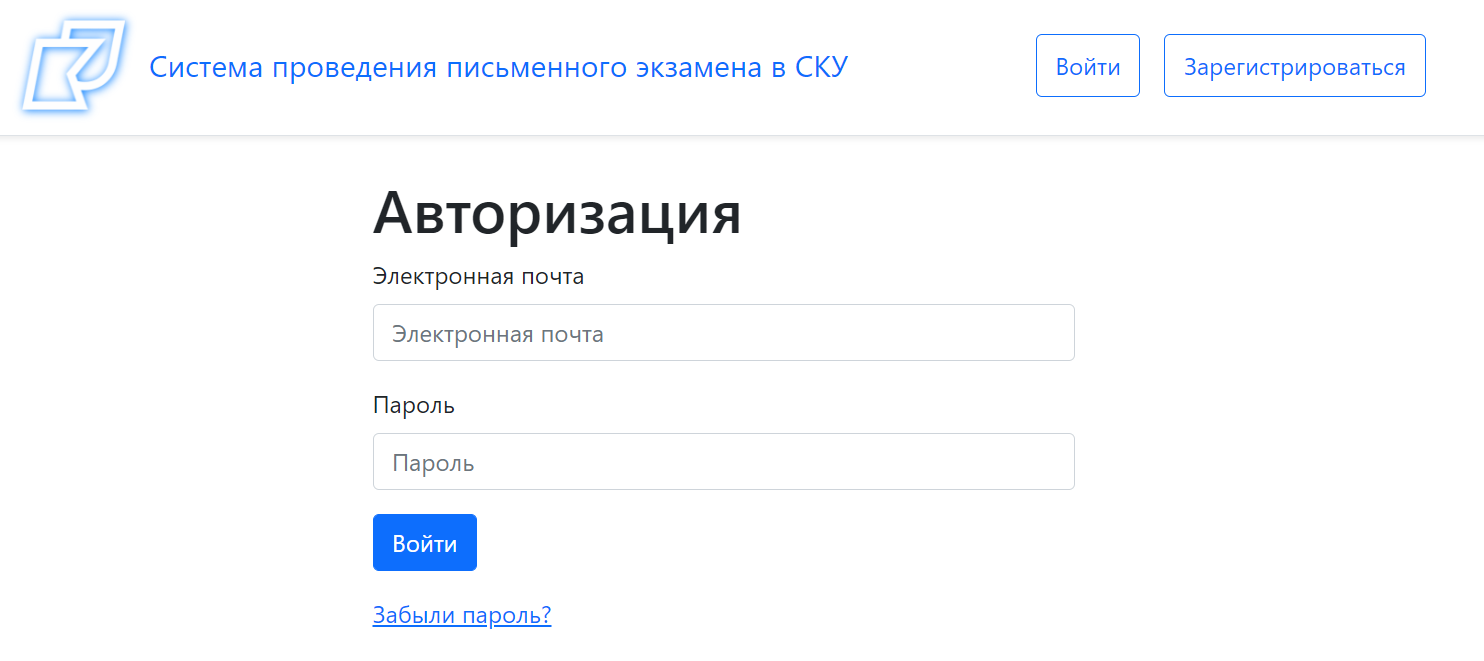


Рисунок 2.12 Авторизация

При необходимости пользователь может восстановить свой пароль, отправив письмо на почту. Восстановление пароля представлено на рисунке 2.13.

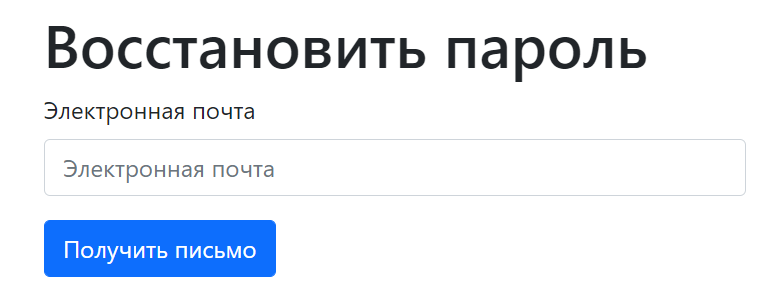


Рисунок 2.13 Восстановление пароля

При нажатии кнопки «Зарегистрироваться» открывается новое окно регистрации с полями ФИО, электронной почты и пароля. При регистрации пользователь выбирает свой статус: студент, преподаватель или заведующий кафедрой. Регистрация представлена на рисунке 2.14.

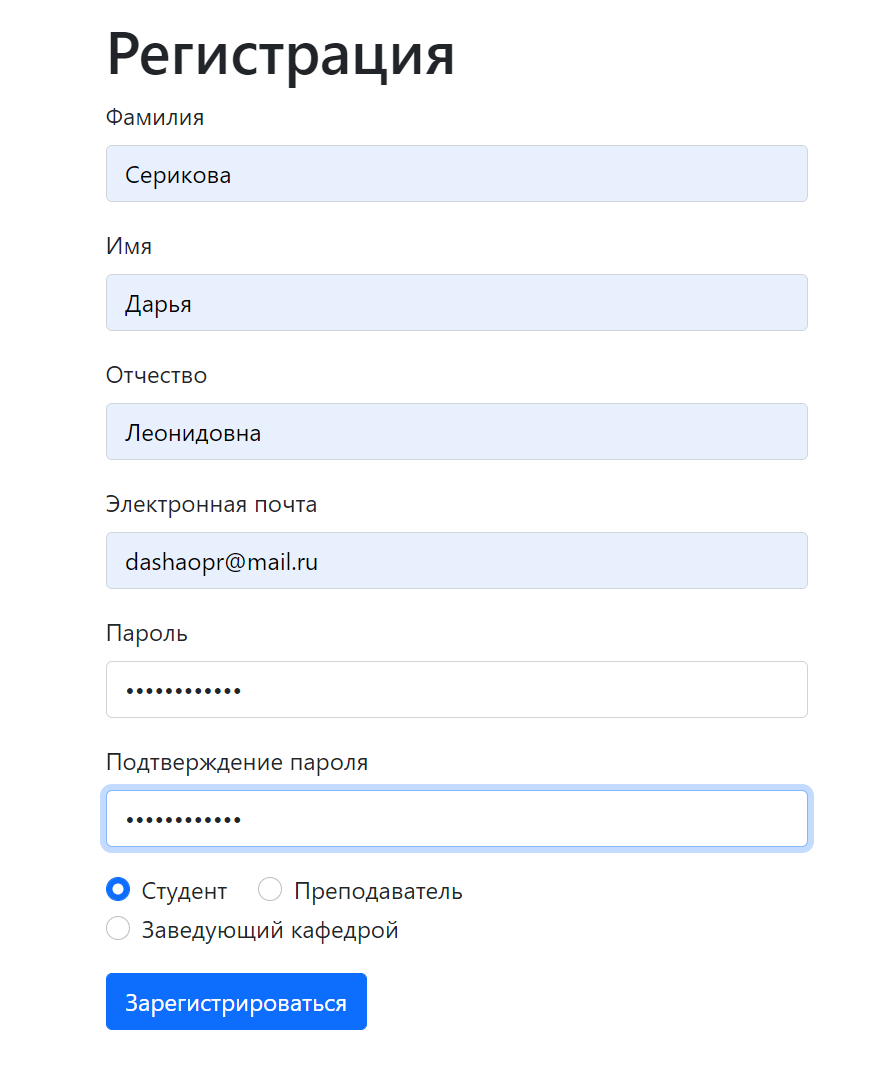


Рисунок 2.14 Регистрация

В нижней части каждой страницы находится подвал. Он состоит из описания разработчиков проекта. Подвал WEB-сайта представлен в соответствии с рисунком 2.15.

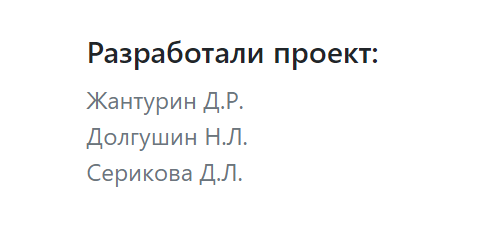


Рисунок 2.15 Подвал сайта

После входа преподаватель может прийти по вкладке «Экзамены» и создать экзамен. Для создания экзамена необходимо выбрать дисциплину из присутствующих, выбрать группу, дату и время экзамена, добавить билеты и критерии. Создание экзамена представлено на рисунке 2.16.

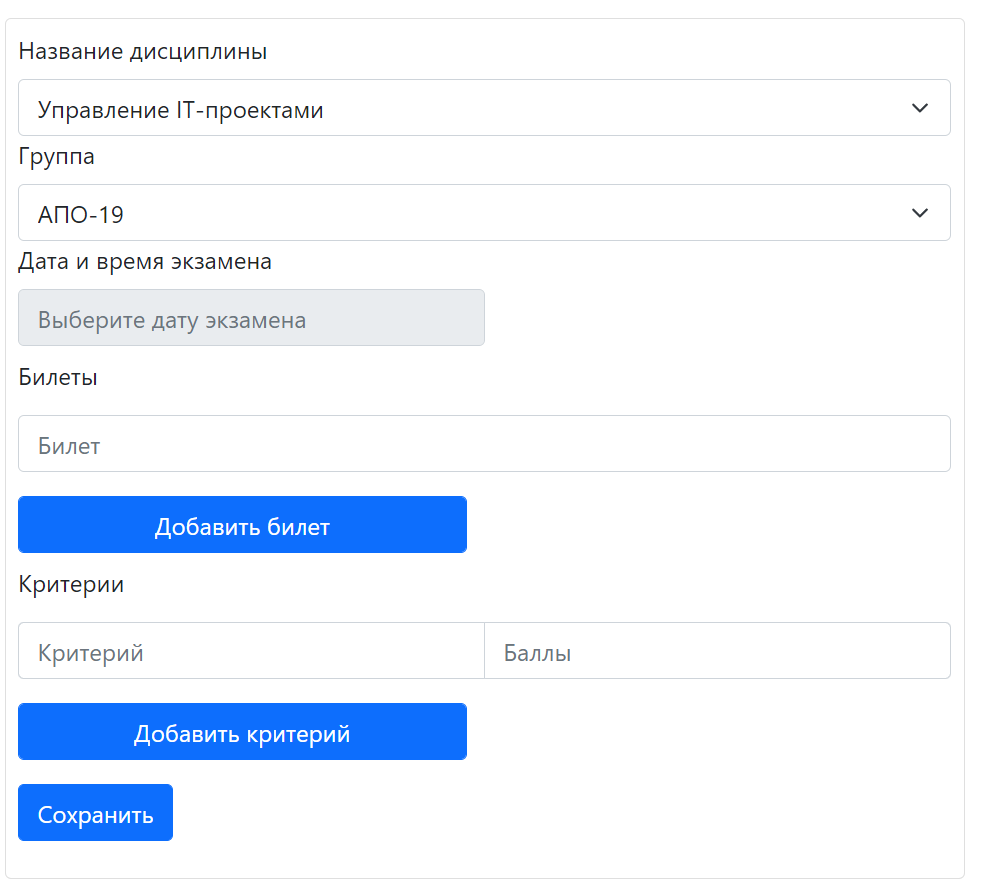


Рисунок 2.16 Создание экзамена

Преподаватель выбирает одну из дисциплин, которые он преподает в университете. Выбор дисциплин представлен на рисунке 2.17.

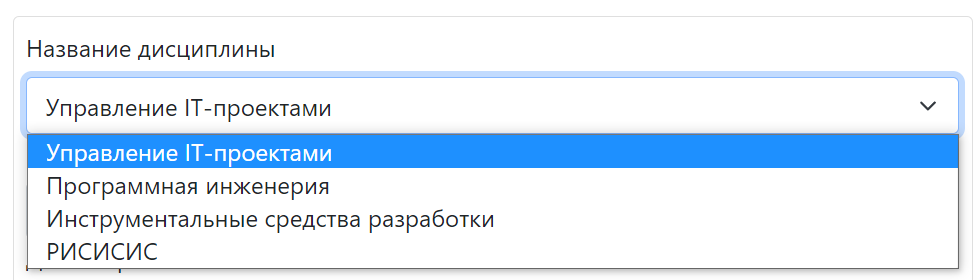


Рисунок 2.17 Выбор дисциплин

Выбор даты и времени представлен на рисунке 2.18.

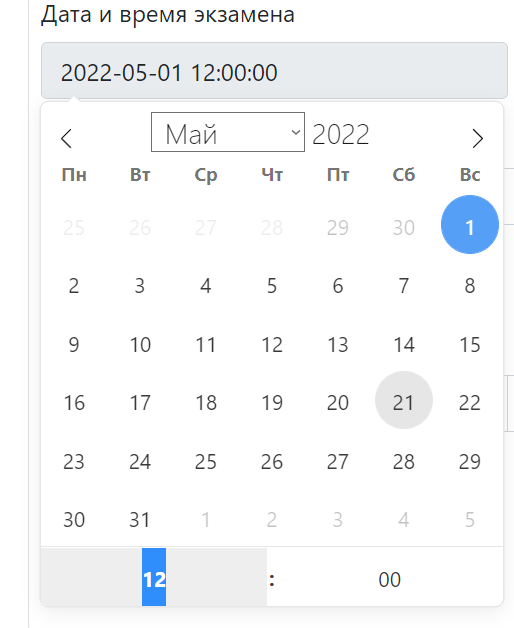


Рисунок 2.18 Выбор даты и времени экзамена

После создания экзамена преподавателем его должен утвердить заведующий кафедрой. Функционал заведующего кафедры представлен на рисунке 2.19. Страница утверждения экзаменов представлена на рисунке 2.20.

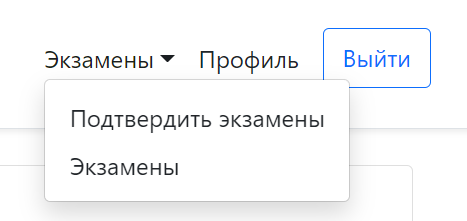


Рисунок 2.19 Функционал заведующего кафедры

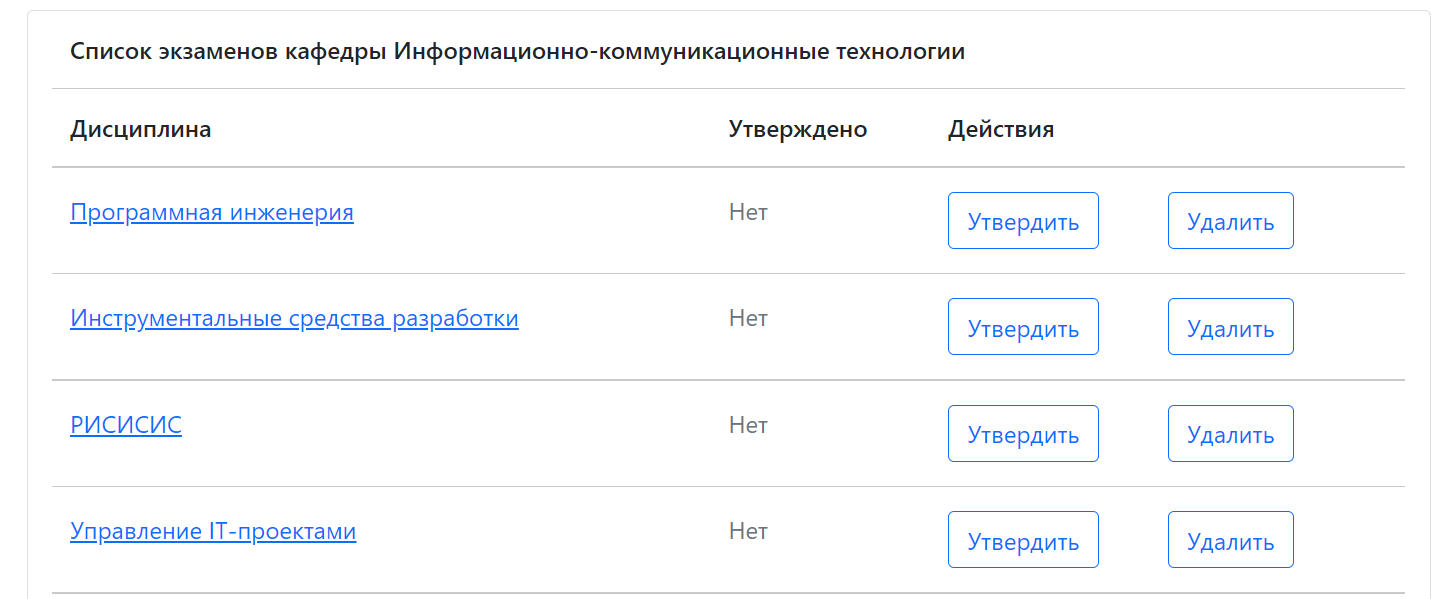


Рисунок 2.20 Страница утверждения экзаменов

После нажатия кнопки «Утвердить» статус экзамена меняется с «Нет» на «Да». Утвержденный экзамен по дисциплине «Программная инженерия» представлен на рисунке 2.21.

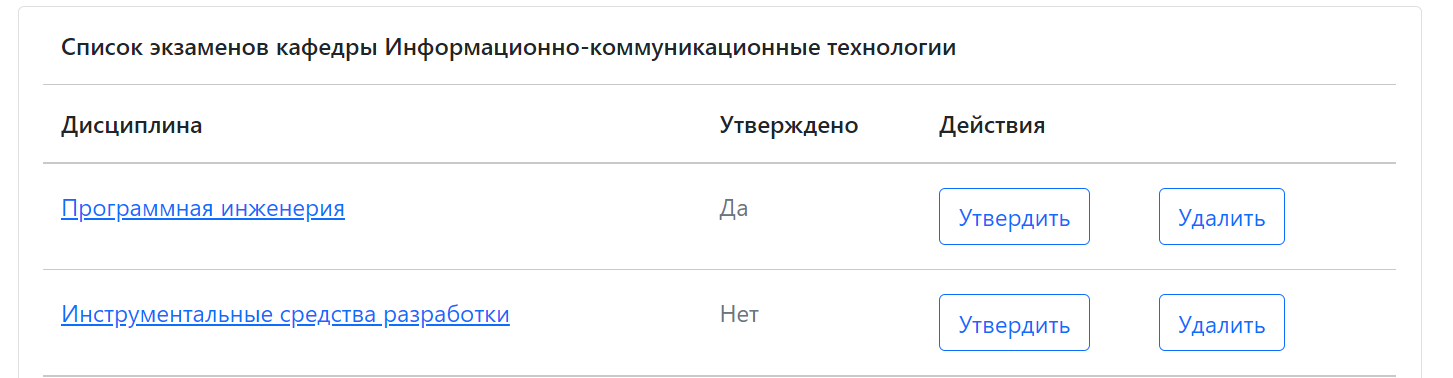


Рисунок 2.21 Утвержденный экзамен

В профиле преподавателя и заведующего указывается кафедра и факультет, к которым он относится. Профиль пользователя представлен на рисунке 2.22.

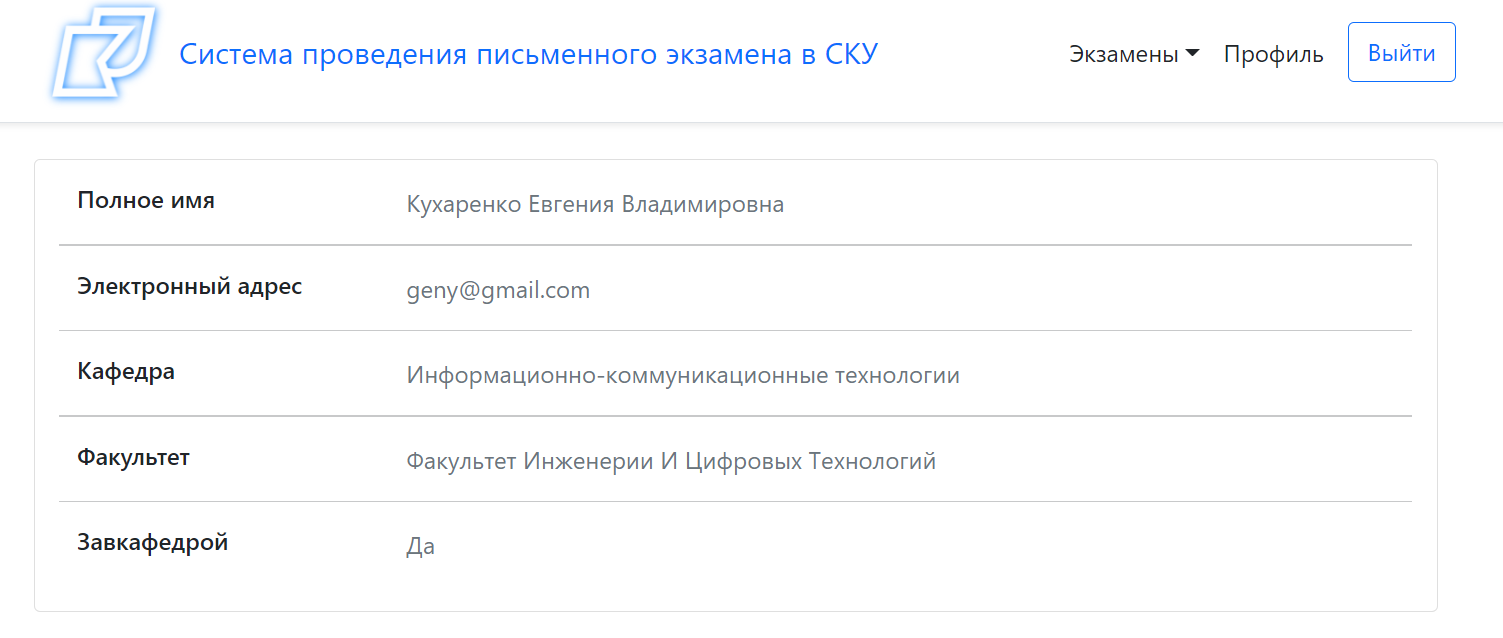


Рисунок 2.22 Профиль пользователя

При прохождении экзамена студенту случайным образом выпадает билет и критерии к нему. Студент пишет свой ответ в поле ответа или прикрепляет файл и отправляет его преподавателю. Список назначенных экзаменов представлен на рисунке 2.23.

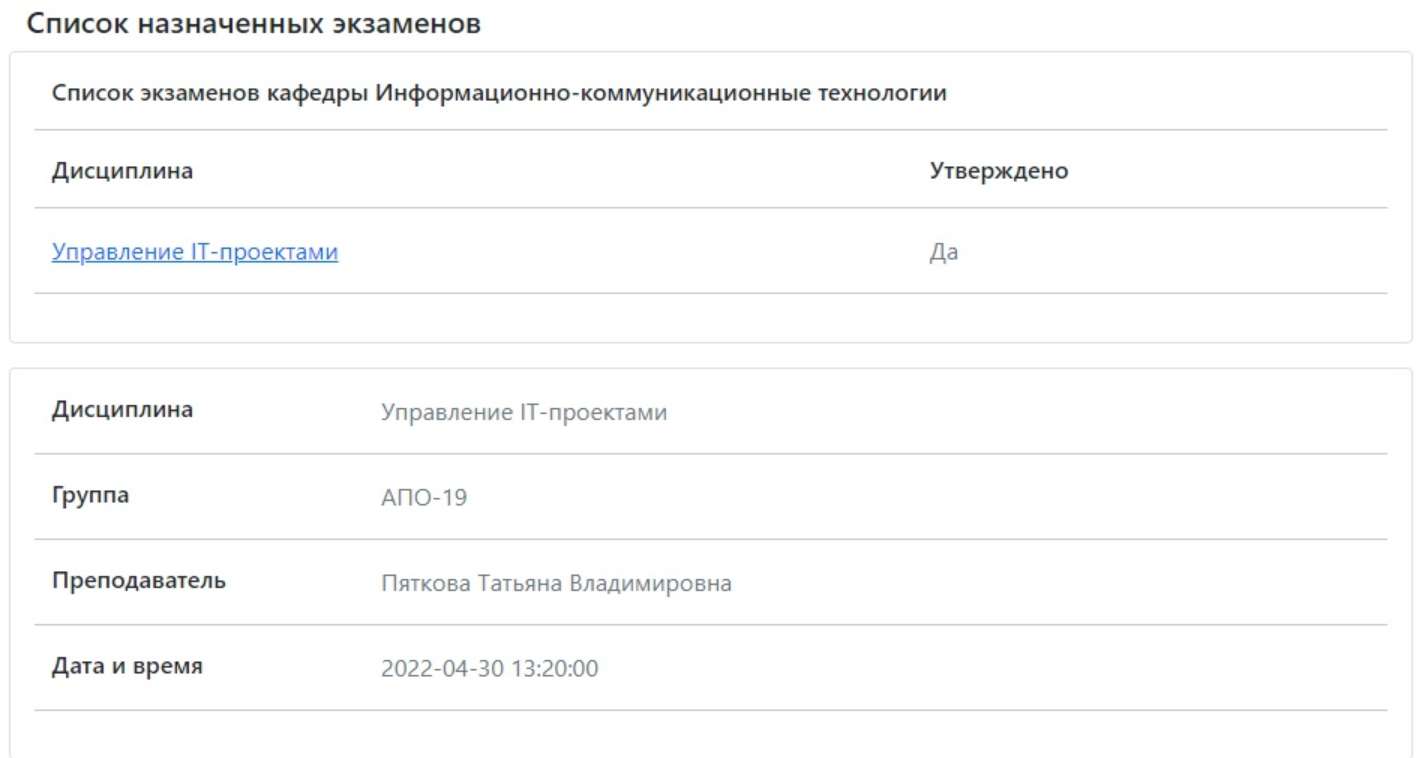


Рисунок 2.23 Список назначенных экзаменов

Прохождение экзамена представлено на рисунке 2.24.

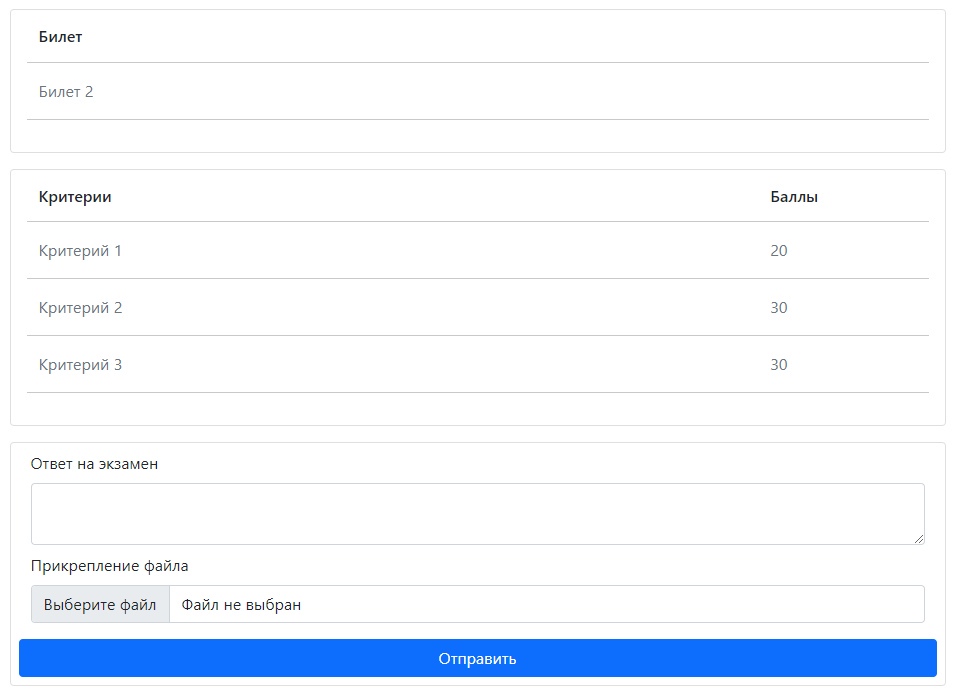


Рисунок 2.24 Прохождение экзамена

После прохождения экзамена студентом преподаватель может просмотреть его ответ и выставить оценку. Выставление оценки представлено на рисунке 2.25.

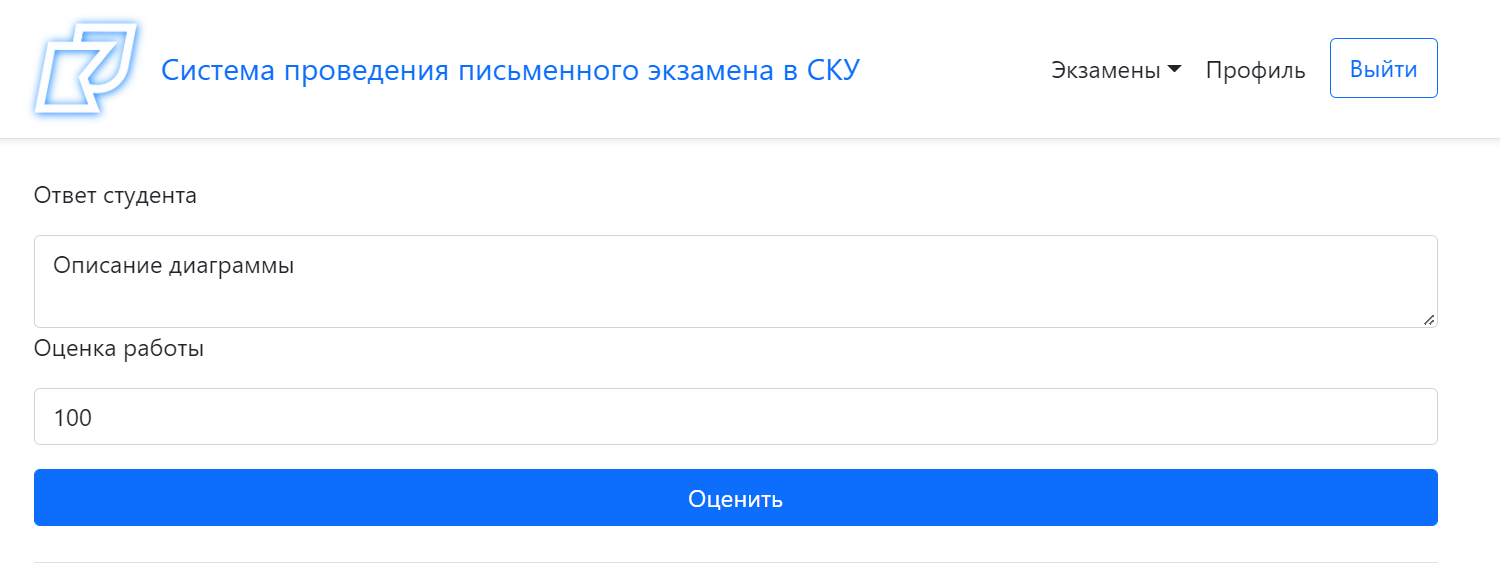


Рисунок 2.25 Выставление оценки

На веб-сайте была создана панель администратора. Вход в панель администратора представлен на рисунке 2.26.

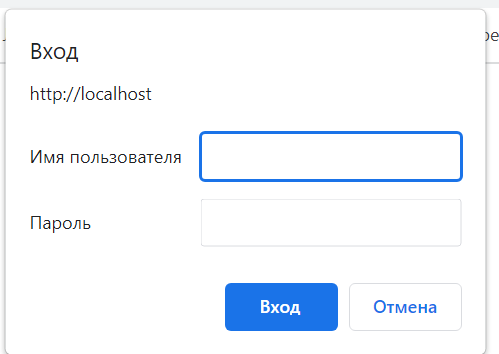


Рисунок 2.26 Вход в панель администратора

Панель администратора включает в себя разделы «Пользователи», «Дисциплины», «Кафедры». Панель администратора представлена на рисунке 2.27.

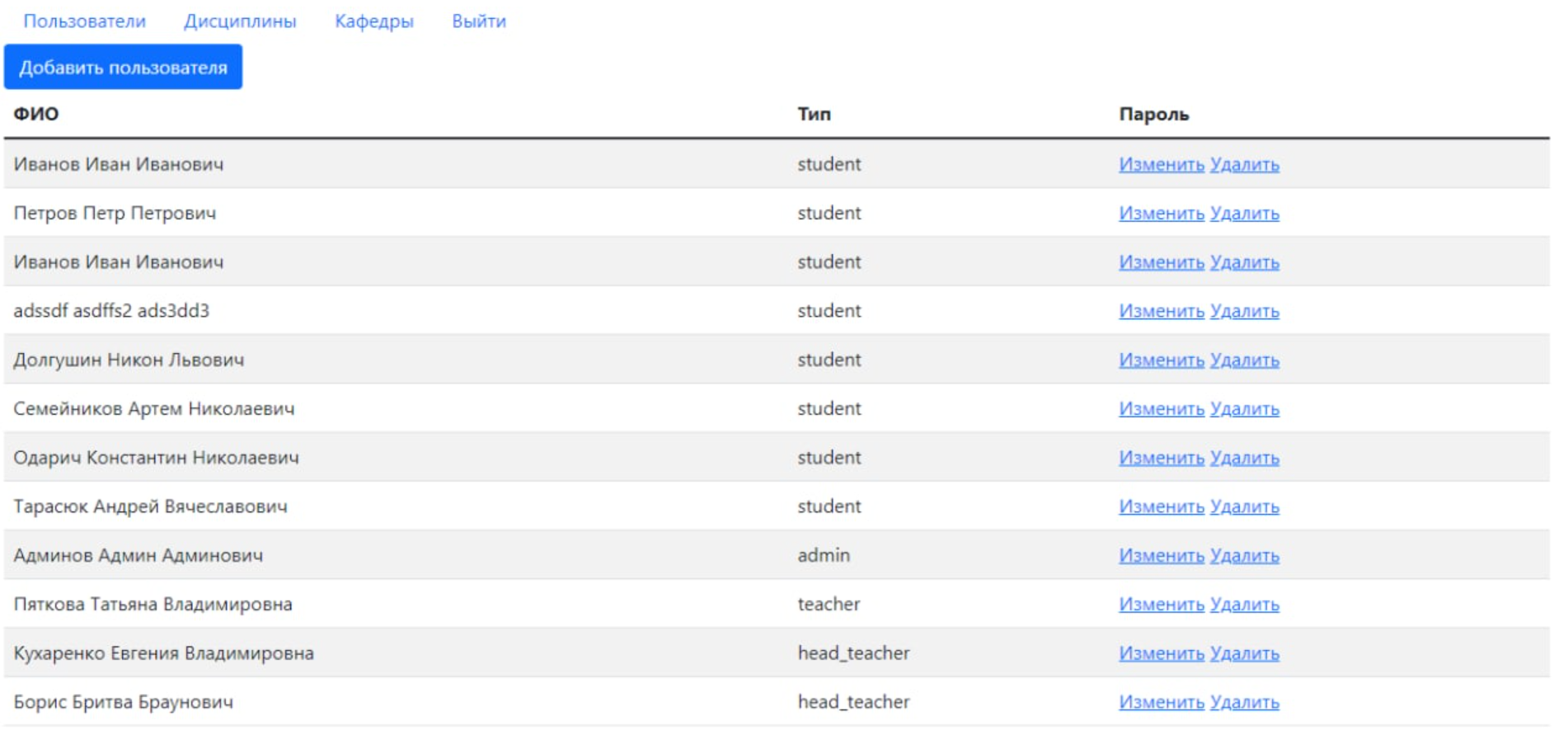


Рисунок 2.27 Панель администратора

Перейдя по кнопке «Пользователи» администратор может добавить нового пользователя. Добавление нового пользователя представлено на рисунке 2.28.

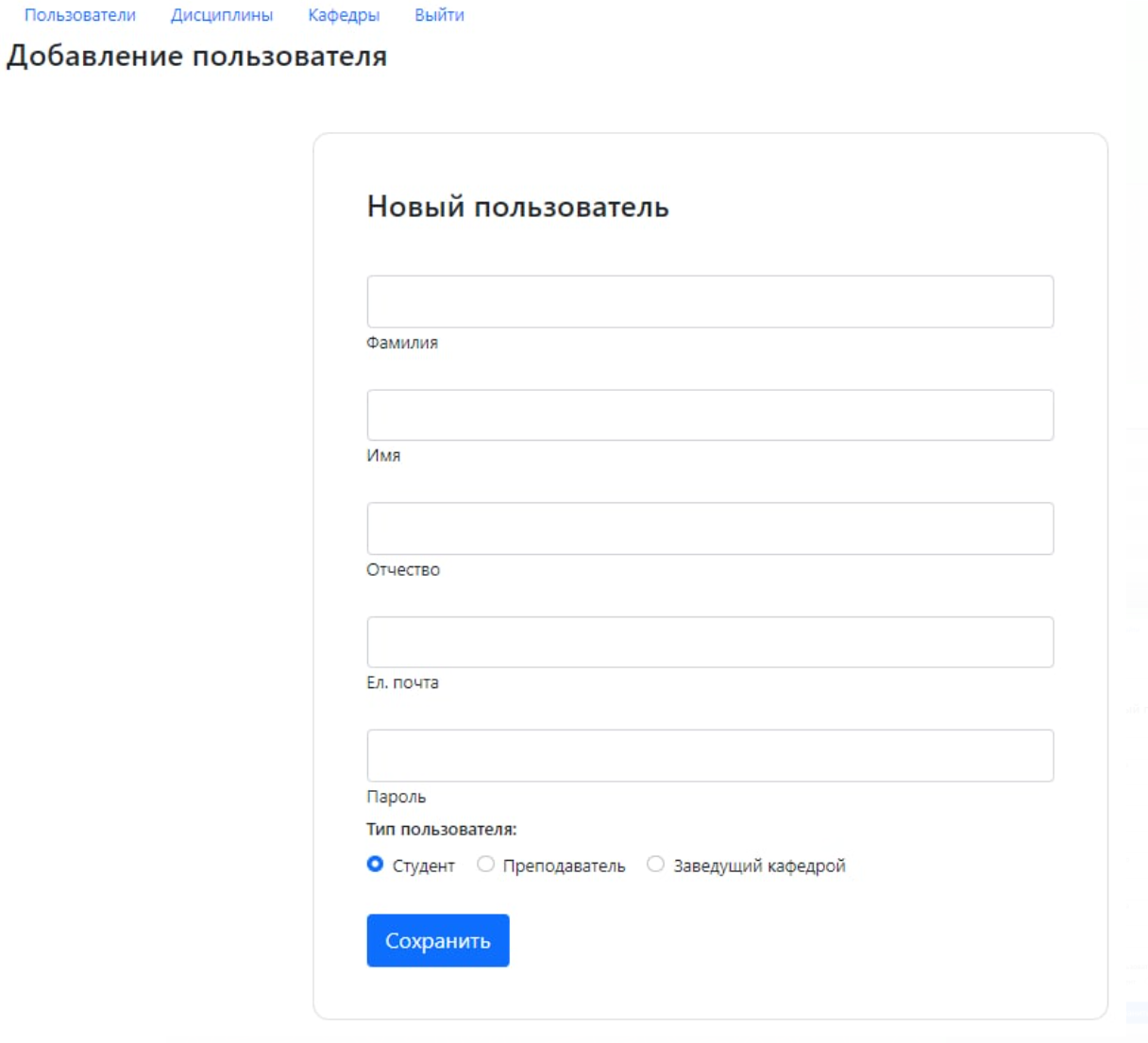


Рисунок 2.28 Добавление пользователя

Таким образом, был завершен пятый спринт. Результатами последнего спринта являются отформатированный отчет, описанное тестирование проекта, заключение и список использованной литературы.

# ЗАКЛЮЧЕНИЕ

В ходе выполнения семестровой работы была разработана система, позволяющая контролировать знания студентов онлайн. Для решения данной проблемы была создана система для проведения «письменного» экзамена, предоставляющая дополнительный функционал.

Данная система помогает избежать возможность ошибки при кодировании и декодировании логина студента, возможность нарушения академической честности, большие затраты по рабочему времени, трудовых и материальных ресурсов на ведение и контроль документов, неизбежно большое количество ошибок и описок при проведении выборки необходимых сведений и подготовке данных к различным отчетам.

Система была разработана по методологии Scrum, которая включает в себя планирование ресурсов и составление графика работ. Scrum обладает достаточно привлекательными достоинствами. Scrum ориентирован на клиента, адаптивен. Scrum дает клиенту возможность делать изменения в требованиях в любой момент времени. Возможность изменения требований привлекательна для многих заказчиков ПО.

Управление различными процессами при помощи компьютера позволяет добиться более высокой производительности труда и сэкономить массу времени. Высококачественная автоматизация технологических процессов значительно облегчает работу университета в целом.

Для разработки программного средства был проведен анализ предметной области, выбраны средства проектирования базы данных, программного средства. В результате реализации семестрового проекта были выполнены поставленные цели и задачи. Ресурс имеет четкую логическую структуру и удобную навигацию. Вся информация, находящаяся на WEB-сайте, размещается в базе данных MySQL.Для разработки использовались такие технологии как:

— CSS;

— HTML;

— PHP;

— MySQL.

На данный момент WEB-сайт предоставляет следующий функционал:

— Авторизация;

— Добавление билетов и критериев;

— Утверждение билетов и критериев;

— Добавление ответа на билет;

— Выставление оценки;

— Просмотр оценки.

На данном этапе развития функционал разработанной системы логически завершён. В случае же продолжения разработки в сторону расширения функционала следует добавить возможность пересдачи экзамена. Исходный код страниц структурирован, что позволяет без труда изменить или дополнить часть его функционала человеку, обладающему базовыми знаниями языка программирования PHP.

Таким образом, разработанная в ходе семестрового проекта система для проведения «письменного» экзамена в СКУ упростит задачу проведения экзаменов в университете. Система может быть использована в практической деятельности университета.

# СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. [Электронный ресурс]. – Режим доступа: URL: https://www.youtube.com/watch?v=Rke\_Z1-nvUM. (05.04.2022);
2. [Электронный ресурс]. – Режим доступа: URL: https://www.youtube.com/watch?v=hyUBMmL0WtA. (05.04.2022);
3. [Электронный ресурс]. – Режим доступа: URL: https://www.youtube.com/watch?v=fYFiQ7lpfiE. (05.04.2022);
4. [Электронный ресурс]. – Режим доступа: URL: https://www.youtube.com/watch?v=JfpCicDUMKc. (05.04.2022);
5. [Электронный ресурс]. – Режим доступа: URL: https://www.youtube.com/watch?v=1xkWYCJaBAU. (05.04.2022);
6. [Электронный ресурс]. – Режим доступа: URL: https://www.youtube.com/watch?v=NijFSs03Pd4. (05.04.2022);
7. [Электронный ресурс]. – Режим доступа: URL: .https://www.youtube.com/watch?v=Anz0ArcQ5kI&list=RDCMUCzyuZJ8zZ-Lhfnz41DG5qLw&index=3 (05.04.2022);
8. [Электронный ресурс]. – Режим доступа: URL: https://artisan-team.ru/seo-wiki/toshnota-teksta/ (06.04.2022);
9. Мальчук, Е. HTML и CSS [Текст]: самоучитель / Е. Мальчук. - СПб.: Академия, 2010;
10. 3. Ломов, А. HTML, CSS, СКРИПТЫ. Практика создания САЙТОВ [Текст] / А. Ломов. - СПБ.: БХВ, 2010;
11. 4. Мержевич, В. В. HTML и CSS на примерах [Текст] / В. В. Мержевич. - СПб.: БХВ - Петербург, 2010;
12. 5. Нильсен, Я. Web-дизайн: удобство использования Web-сайтов [Текст] / Я. Нильсен, Х. Лоранжер. - М.: Вильямс, 2010;
13. 6. Роббинс, Д. Web-дизайн [Текст]: справочник / Д. Роббинс. - М.: КУДИЦ - Пресс, 2010;
14. 7. Евсеев, Д. А. Web-дизайн в примерах и задачах [Text]: учебное пособие / Д. А. Евсеев, В. В. Трофимов; ред. В. В. Трофимов. - М.: КНОРУС, 2010;
15. 8. Хольцшлаг, М. Языки HTML и CSS для создания Web-сайтов [Текст]: учебный курс / М. Хольцшлаг. - М.: Триумф, 2010.