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](#_Toc98280509)

[1 Аналитическая часть 5](#_Toc98280510)

[1.1 Исследование предметной области 5](#_Toc98280511)

[2 Проектирование автоматизированной информационной системы 7](#_Toc98280512)

[2.1 Распределение задач 7](#_Toc98280513)

[2.2 Описание баз данных 11](#_Toc98280514)

[2.3 Разработка архитектуры программного средства 13](#_Toc98280515)

[2.4 Описание используемых алгоритмов 14](#_Toc98280516)

[2.5 Тестирование программного средства 20](#_Toc98280517)

[ЗАКЛЮЧЕНИЕ 24](#_Toc98280518)

[СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ 26](#_Toc98280519)

# ВВЕДЕНИЕ

На сегодняшний день автоматизация деятельности становится неотъемлемой частью практически любого предприятия, поэтому рассмотрение *проблемы неудобства проведения «письменного» экзамена является актуальным.* *Автоматизация бизнес-процесса проведения экзамена является актуальной задачей* и способствует оптимизации работы организации в целом. 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.

Рассматривая проблему автоматизации проведения экзамена, стоит отметить, что предпосылками автоматизации 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.

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.

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.

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». Из этого следует, что в современное время есть потребность в организации онлайн-экзаменов.

*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. Данная 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.

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.

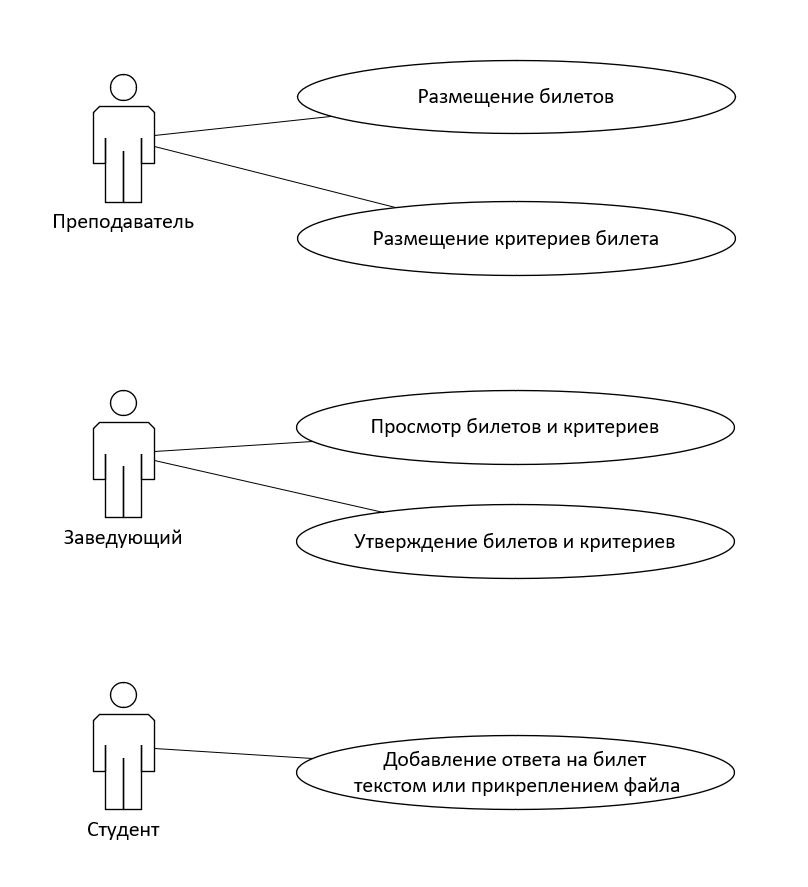


Рисунок 1 Диаграмма вариантов использования классов системы

Диаграмма вариантов использования классов системы представлена на рисунке 1.

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. Бизнес-процессы системы для проведения «письменного» экзамена представлены на рисунке 2.

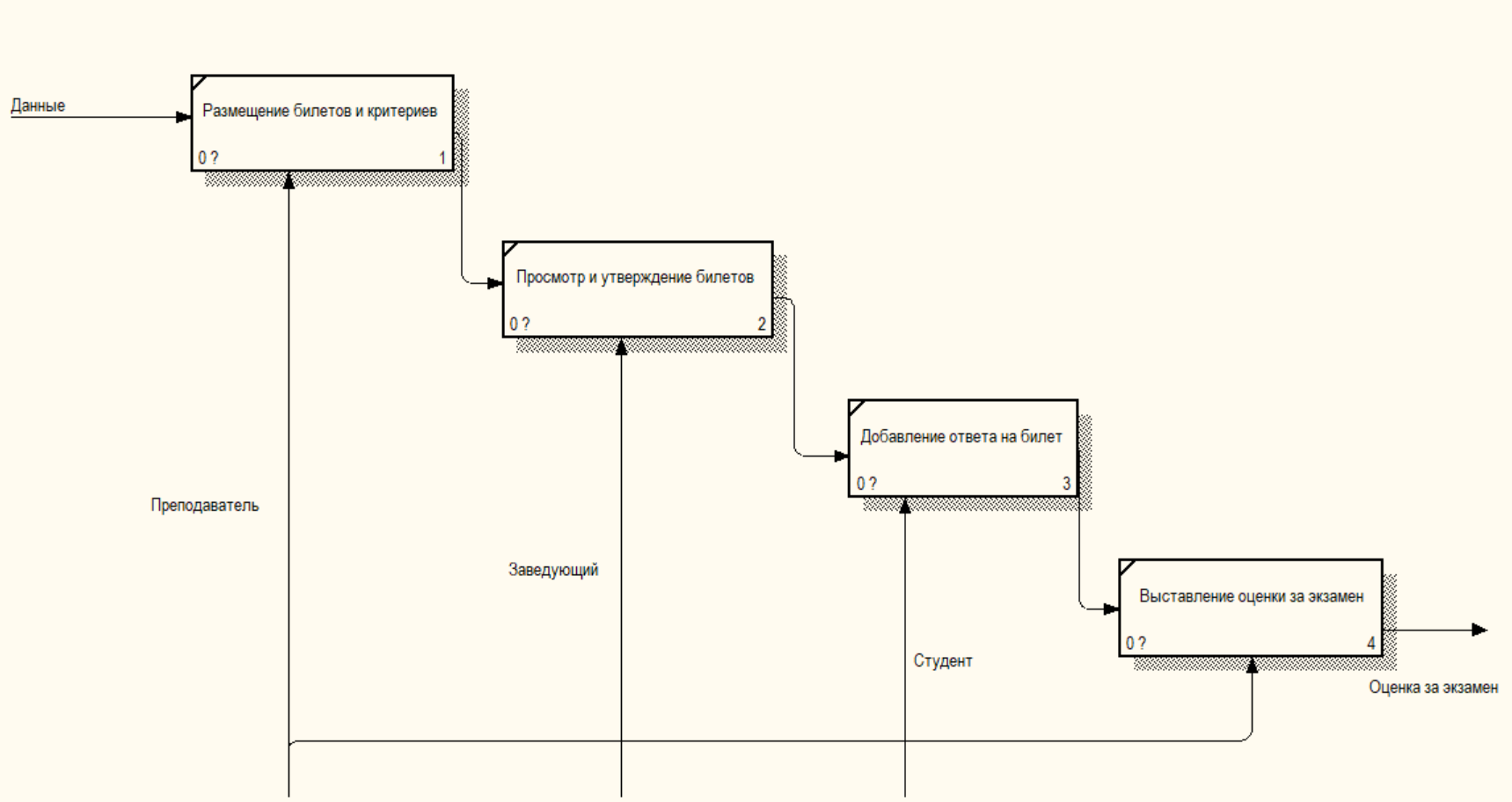


Рисунок 2 Бизнес-процессы системы

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.

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 автоматизированной информационной системы

## 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, создание EXE-файла. 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 | Создание формы для добавления ответа на билет | Описание списка использованной литературы | Создание EXE-файла |
| Долгушин Никон | Создание макетов проекта, реализация структуры MVC | Описание алгоритмов | Создание формы входа | 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 | Создание формы для просмотра и утверждения билетов | Подключение БД проекта | 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.

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 архитектуры проекта | Добавление формы ответов в раздел тестирования | Описание списка использованной литературы | Добавление в отчет заключения |
| Долгушин Никон | Описание алгоритмов проекта | Описание диаграмм баз данных | Добавление формы входа в раздел тестирования | Добавление формы для размещения билетов в раздел тестирования | Форматирование отчета |
| Жантурин Даниял | Описание бизнес-процессов и архитектуры проекта | Описание баз данных проекта | Добавление формы для утверждения билетов в раздел тестирования | Описание подключенных БД | Описание тестирования проектного средства |

После распределения работ 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

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.

Соответствуя структуре университета на рисунке 3 представлены базы данных 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.

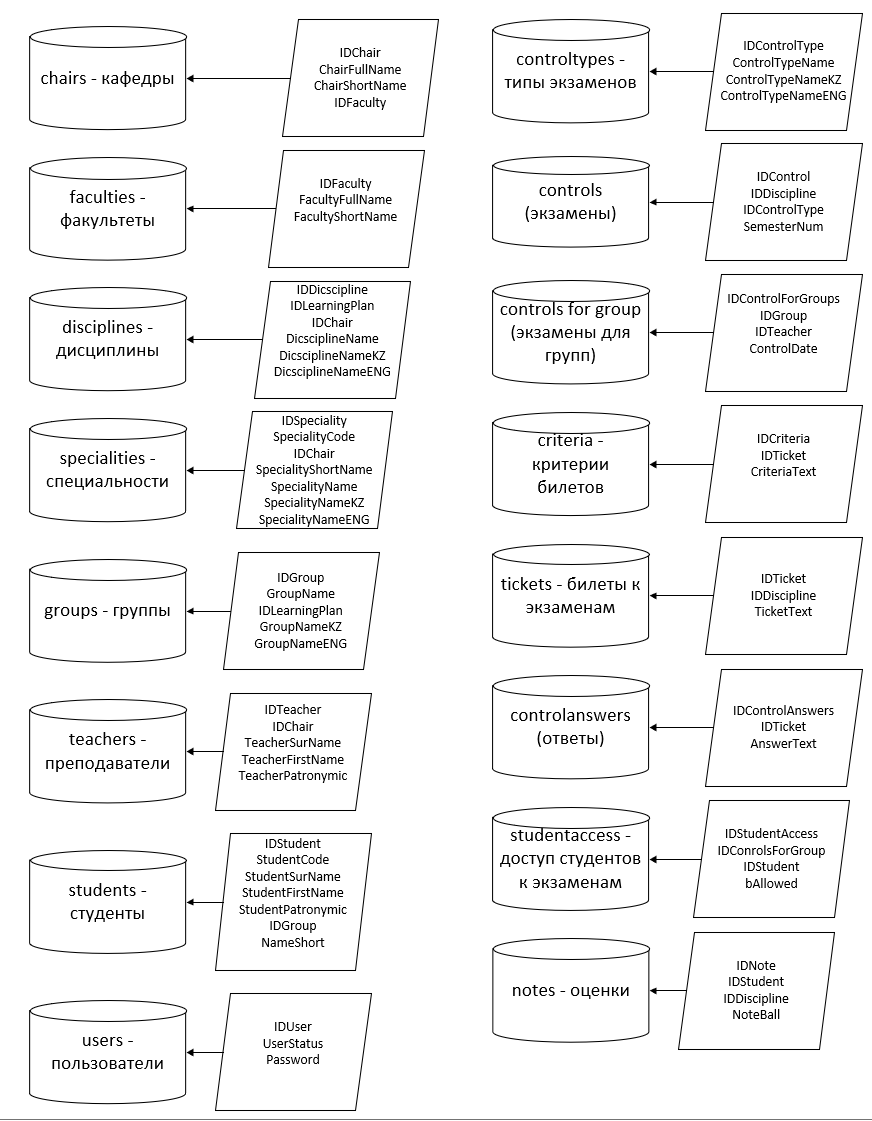


Рисунок 3 Диаграмма баз данных

Диаграмма баз данных представлена в соответствии с рисунком 3. 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.

## 2.3 Разработка архитектуры программного средства

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.

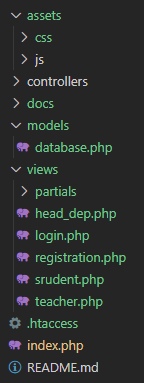


Рисунок 4 Архитектура проекта

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.

На рисунке 5 представлена архитектура проекта, где 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. Представлением является ответ студента на билет и выставленная оценка. Контроллером будут преподаватель и заведующий.

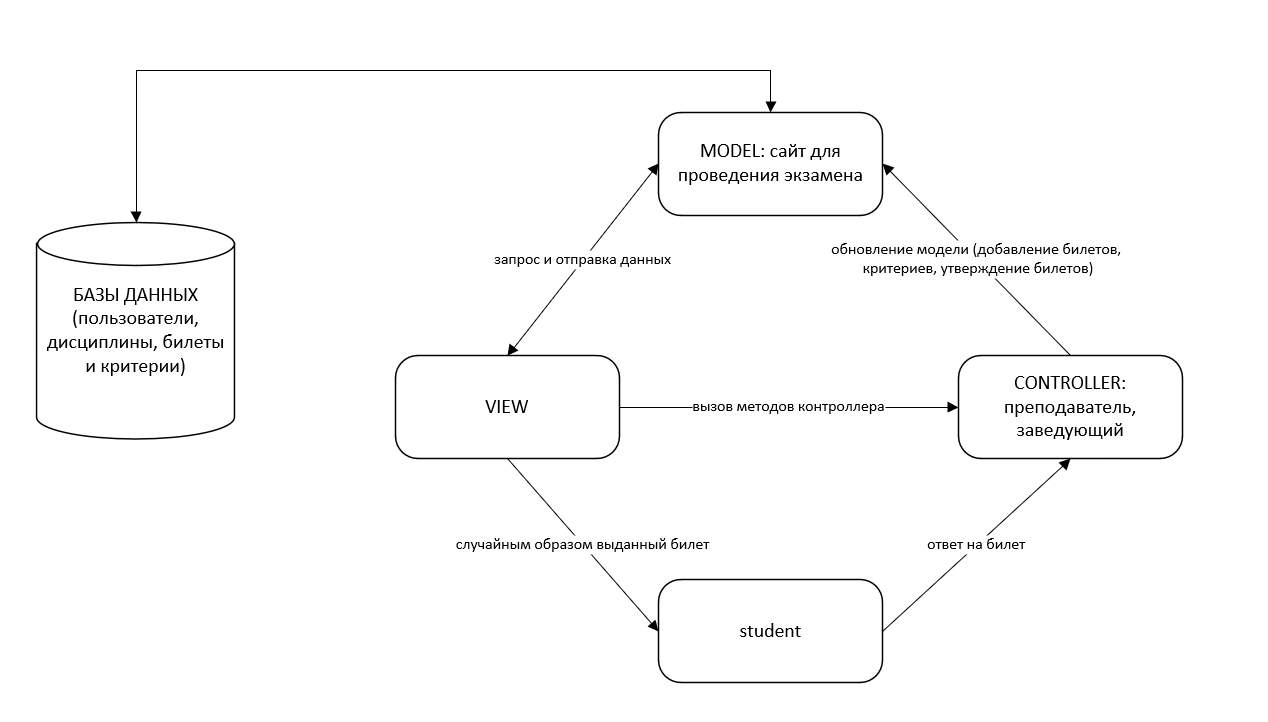


Рисунок 5 Архитектура проекта MVC

## 2.4 Описание используемых алгоритмов

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 6.

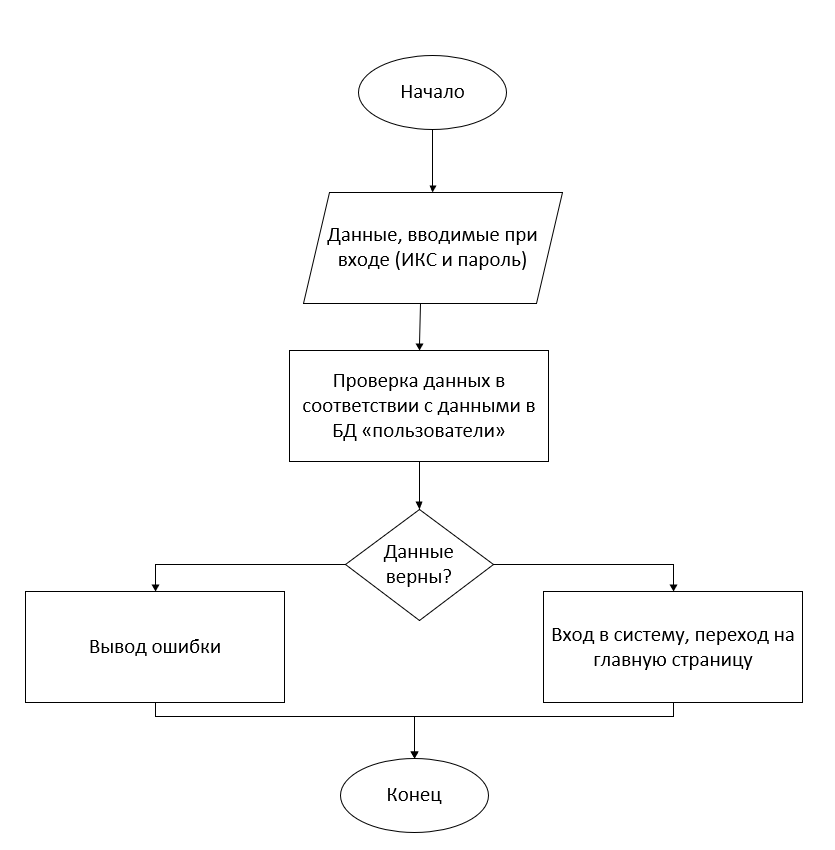


Рисунок 6 Блок-схема входа в систему

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 7.

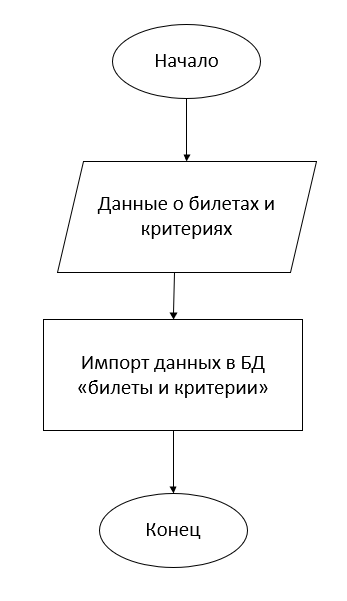


Рисунок 7 Алгоритм добавления билетов и критериев

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 8.

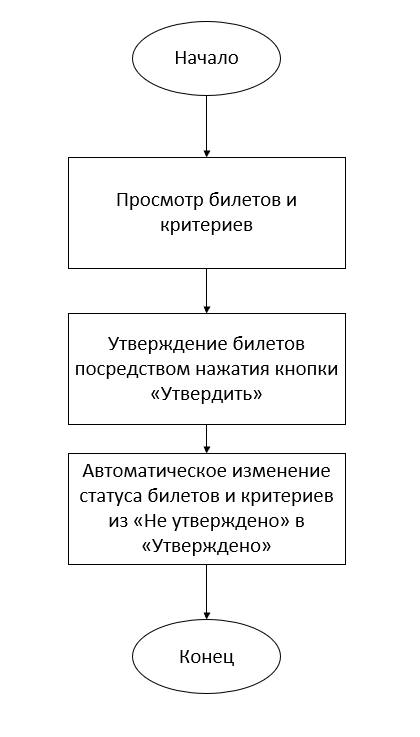


Рисунок 8 Алгоритм утверждения билетов

Последним 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 регистрации посредством блок-схем. Блок-схема алгоритма добавления ответа представлена в соответствии с рисунком 9.

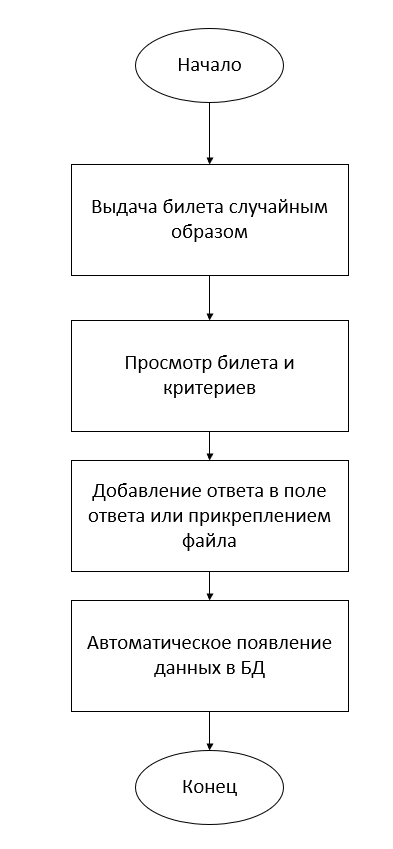


Рисунок 9 Добавление ответа на билет

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

## 2.5 Тестирование программного средства

После сбора архитектуры и создания EXE-файла 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 страница входа с функционалом заполнения ИКС и пароля, дальнейшего входа по ним.

Страница входа представлена на рисунке 10.

Рисунок 10 Страница входа

При нажатии кнопки «Регистрация» открывается новое окно регистрации с полями логина, пароля, использующихся в дальнейшем для входа, а также личные и паспортные данные.

Рисунок 11 Окно регистрации пользователя

Заполнение данных в csv-файл names представлено на рисунке 12.

Рисунок 12 Содержание файла names.csv

После входа или завершения регистрации пользователь переходит на главную страницу, представленную на рисунке 13. Данная страница содержит меню с каталогом товаров, информации о товарах и справку.

Рисунок 13 Главная страница пользователя

Каталог товаров представляет собой страницу с девятью товарами магазина. Каждый товар имеет цену, наименование и кнопку «Добавить в корзину». После добавления товаров в корзину можно перейти в нее по кнопке внизу страницы. Каталог товаров представлен на рисунках 14, 15.

Рисунок 14 Каталог товаров. Часть 1

Рисунок 15 Каталог товаров. Часть 2

При нажатии кнопки «Информация о товарах» выходит меню с выбором «Просмотреть корзину» и «Мои заказы». Меню страницы пользователя представлено на рисунке 16.

Рисунок 16 Меню на странице пользователя

Корзина представляет собой таблицу с номером заказа, наименованием товара, ценой, количеством и общей стоимостью. Товар можно удалить из корзины. Корзина представлена на рисунке 17.

Рисунок 17 Корзина с товарами

При переходе по кнопке «Оформить заказ» открывается страница заказов, представленная на рисунке 18.

Рисунок 18 Страница заказов

На странице заказов можно просмотреть оформленные заказы, а также оставить отзыв о приложении, представленный на рисунке 19.

Рисунок 19 Отзыв о приложении

Справка в меню, представленном на рисунке 20, содержит информацию о программе и разработчиках, представленную на страницах 21, 22 соответственно.

Рисунок 20 Меню справки

Рисунок 21 Справка о программе

Рисунок 22 Справка о разработчиках

Главная страница, представленная на странице 23, администратора включает в себя функцию просмотра обработки данных.

Рисунок 23 Главная страница администратора

Первым алгоритмом обработки данных, представленным на рисунке 24, является вывод статистики товаров с наибольшим спросом за выбранный месяц. Алгоритм выбирает первые 5 товаров с наибольшим спросом и располагает в порядке убывания. Статистика представлена на рисунках 25, 26.

Рисунок 24 Меню страницы администратора для первого запроса

Рисунок 25 Статистика товаров с наибольшим спросом за октябрь

Рисунок 26 Статистика товаров с наибольшим спросом за июнь

Вторым алгоритмом является динамика смен цены заданного товара, представленная на рисунках 27, 28. При наведении на определенный месяц можно выбрать его и просмотреть конкретную статистику.

Рисунок 27 Меню страницы администратора для второго запроса

Рисунок 28 Динамика изменения цен товаров за 2021 год

Третий алгоритм статистики улиц проживания абонентов представлен на рисунках 29, 30 в виде круговой диаграммы.

Рисунок 29 Меню страницы администратора для третьего запроса

Рисунок 30 Статистика улиц доставки товаров

Четвертый и последний алгоритм стоимости товара по декадам представлен на рисунках 31, 32.

Рисунок 31 Меню страницы администратора для четвертого запроса

Рисунок 32 Статистика стоимости товаров за 6 декаду

# ЗАКЛЮЧЕНИЕ

В ходе выполнения семестровой работы была решена проблема автоматизации деятельности, что является неотъемлемой частью практически любого предприятия. Для решения данной проблемы была создана автоматизированная информационная система для магазина Wildberries, предоставляющая дополнительный функционал. А именно использование данного приложения позволяет пользователю:

1. Зарегистрироваться в приложении «Wildberries» для оформления дальнейших покупок.

2. Просмотреть каталог товаров и добавить товары в корзину.

3. Просмотреть корзину, оформить заказ товаров, находящихся в корзине, либо удалить товары из корзины.

4. Просмотреть информацию об оформленных заказах в разделе «Мои заказы».

Автоматизированная информационная система также включает в себя функционал для администратора, включающий в себя следующую обработку данных:

1. Просмотр данных о пользователях и товарах в подключенных БД;

2. Просмотр статистики списка товаров, пользующихся наибольшим спросом (максимальное количество позиций заказов) у населения за заданный месяц;

3. Просмотр динамики изменения стоимости заданного товара за заданный период по месяцам;

4. Просмотр статистики списка наименований улиц, на которых проживают абоненты предприятия;

5. Просмотр статистики стоимости товара по декадам.

Автоматизированная информационная система проста в установке, при этом в случае отсутствия одного или сразу двух из модулей зависимости система будет функционировать в рамках имеющихся возможностей, не приводя к критическим ошибкам.

В соответствии с концепцией повторного использования кода, являющейся одной из основ Open Source идеологии, логика модуля опирается на ранее существовавший – хорошо отлаженный и проверенный временем функционал двух других модулей, гармонично взаимодействуя с ним – тем самым расширяя возможности пользователя вообще и веб-мастера в частности в области удобной и понятной организации контента в приложении.

На данном этапе развития функционал автоматизированной информационной системы логически завершён. В случае же продолжения разработки в сторону расширения функционала следует добавить возможность большего переопределения исходных настроек администраторам сайта. Исходный код модуля регистрации пользователя подробно прокомментирован, что позволяет без труда изменить или дополнить часть его функционала человеку, обладающему базовыми знаниями языка программирования Python.

Таким образом, предприятию «Wildberries» отныне не придется совершать большие затраты рабочего времени, трудовых и материальных ресурсов на ведение и контроль документов, поддержание данных в достоверном состоянии. Автоматизированная информационная система помогает избежать большое количество ошибок при проведении выборки необходимых сведений и подготовке данных к различным отчетам.

# СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. [Электронный ресурс]. – Режим доступа: URL: https://www.youtube.com/watch?v=Rke\_Z1-nvUM. (23.02.2022)
2. [Электронный ресурс]. – Режим доступа: URL: https://www.youtube.com/watch?v=hyUBMmL0WtA. (23.02.2022)
3. [Электронный ресурс]. – Режим доступа: URL: https://www.youtube.com/watch?v=fYFiQ7lpfiE. (23.02.2022)
4. [Электронный ресурс]. – Режим доступа: URL: https://www.youtube.com/watch?v=JfpCicDUMKc. (23.02.2022)
5. [Электронный ресурс]. – Режим доступа: URL: https://www.youtube.com/watch?v=1xkWYCJaBAU. (23.02.2022)
6. [Электронный ресурс]. – Режим доступа: URL: https://www.youtube.com/watch?v=NijFSs03Pd4. (23.02.2022)
7. [Электронный ресурс]. – Режим доступа: URL: https://cyberleninka.ru/article/n/organizatsiya-onlayn-kontrolya-v-usloviyah-distantsionnogo-obucheniya-sredstvami-tsifrovyh-tehnologiy/viewer. 23.02.2022)
8. Мэтиз Э. Изучаем Python. Программирование игр, визуализация данных, веб-приложения. — СПб.: Питер, 2017. — 496 с.: ил. — (Серия «Библиотека программиста»)
9. Рядченко, В.П. Программирование на языке высокого уровня Python: учебно-методическое пособие / В.П. Рядченко, Л.М. Эльканова, Л.М. Шавтикова. – Черкесск: БИЦ СевКавГГТА, 2018. –144с.
10. Федоров Д.Ю. Основы программирования на примере языка Python: учебное пособие. – СПб.: Питер, 2019. – 152 с.