МИНИСТЕРСТВО ОБРАЗОВАНИЯ ОМСКОЙ ОБЛАСТИ

бюджетное профессиональное образовательное учреждение Омской области

«ОМСКИЙ АВТОТРАНСПОРТНЫЙ КОЛЛЕДЖ»

(БПОУ «Омский АТК»)

**Специальность: 09.02.03 «Программирование в компьютерных системах»**

**КУРСОВОЙ ПРОЕКТ**

**по МДК.01.01 Системное программирование**

**Тема: Проектирование система дистанционного обучения**

Выполнил студент группы ПКС351

Бронзов С.А

Проверил преподаватель

Курчевский В.Е.

Оценка\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Омск 2019

МИНИСТЕРСТВО ОБРАЗОВАНИЯ ОМСКОЙ ОБЛАСТИ

бюджетное профессиональное образовательное учреждение Омской области

«ОМСКИЙ АВТОТРАНСПОРТНЫЙ КОЛЛЕДЖ»

(БПОУ «Омский АТК»)

|  |  |
| --- | --- |
|  | УТВЕРЖДАЮ |
|  | Зам. директора |
|  | \_\_\_\_\_\_\_\_\_\_\_\_\_И.В. Сидоренко |
|  | «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_20\_\_ г. |
|  |  |

##### **ЗАДАНИЕ**

**на курсовой проект**

**по МДК.01.01 Системное программирование**

студенту Бронзову Сергею Андреевичу

группы ПКС351 курса 4

специальность 09.02.03 «Программирование в компьютерных системах»

**Тема: *Проектирование система дистанционного обучения***

**ВВЕДЕНИЕ**

Актуальность заданной темы, цели и задачи работы.

**1 ИССЛЕДОВАТЕЛЬСКИЙ РАЗДЕЛ**

1.1 Основные понятия

1.2 Анализ существующих аналогов

1.3 Техническое задание

**2 ТЕХНОЛОГИЧЕСКИЙ РАЗДЕЛ**

2.1 Назначение разработки

2.2 Требования к проекту

2.3 Выбор программных и технических средств

2.4 Проектирование системы

**3 ОРГАНИЗАЦИОННЫЙ РАЗДЕЛ**

3.1 Организация графика разработки

3.2 Система контроля версий

**ЗАКЛЮЧЕНИЕ**

**СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ**

Дата выдачи задания на курсовой проект «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_г.

Срок выполнения курсового проекта «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_г.

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

Задание рассмотрено и одобрено на заседании ЦМК «Информатика и вычислительная техника. Математика» от «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_г.

Протокол №\_\_\_\_\_\_ Председатель ЦМК \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Харченко В.М./

Задание получил «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_г.

**Студент**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 4](#_Toc5862127)

[1 ИССЛЕДОВАТЕЛЬСКИЙ РАЗДЕЛ 5](#_Toc5862128)

[1.1 Основные понятия 5](#_Toc5862129)

[1.2 Анализ существующих аналогов 6](#_Toc5862130)

[1.3 Техническое задание 10](#_Toc5862131)

[2 ТЕХНОЛОГИЧЕСКИЙ РАЗДЕЛ 11](#_Toc5862132)

[2.1 Назначение разработки 11](#_Toc5862133)

[2.2 Требования к проекту: 12](#_Toc5862134)

[2.3 Выбор программных и технических средств 15](#_Toc5862135)

[2.4 Проектирование системы 16](#_Toc5862136)

[3 ОРГАНИЗАЦИОННЫЙ РАЗДЕЛ 21](#_Toc5862137)

[3.1 Организация графика разработки 21](#_Toc5862138)

[3.2 Система контроля версий 24](#_Toc5862139)

[ЗАКЛЮЧЕНИЕ 26](#_Toc5862140)

[СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ 27](#_Toc5862141)

**ВВЕДЕНИЕ**

В последние десятилетия появилась новая проблема развития образования в России. Знания стареют каждые 3-5 лет, а технологические знания - каждые 2-3 года. Пройдет еще немного времени и это будет 1,5-2 года. Объем знаний выпускников ВУЗов удваивается каждые 3-4 года. Если не менять образовательных технологий, то качество подготовки специалистов будет объективно отставать от требуемого на рынке труда. Усвоение знаний студентами с помощью информационных и коммуникационных технологий по самым нижним оценкам на 40-60% быстрее, или больше, в единицу времени, чем с обычными технологиями.

Учебно-методическая база, образовательно-информационные технологии любого ВУЗа таковы, что  они вообще не зависят от того, дневная  ли это форма обучения, или заочная, или удаленная. Если знания, весь учебный  материал, вся его дидактическая  составляющая оформлены и находятся  в формализованном виде, в компьютерах, то все равно, в принципе, куда подать эти знания: то ли в аудиторию, то ли находящемуся за пределами  учебного заведения пользователю.

Одним из видов  инноваций в организации профессионального  образования является введение дистанционного обучения.

  В последние десятилетия дистанционные образовательные технологии в России получили интенсивное развитие. Министерством образования РФ разработано специальное направление, научно-методическая программа, выделены средства на развитие и становление дистанционного образования. Но получит ли дистанционное обучение в России столь же широкое распространение и применение, как на западе, покажет только время. А пока попробуем проанализировать системы дистанционного обучения действующие в России на данный момент.

Для реализации программы необходимо:

1. Проанализировать существующие аналоги и сделать определенные выводы,
2. Спроектировать графический интерфейс,
3. Составить и разработать Базу Данных для составления курсов и тестов, а так же для их проссмотра,
4. Реализовать задуманное.
5. **ИССЛЕДОВАТЕЛЬСКИЙ РАЗДЕЛ**
   1. **Основные понятия**

Автоматизированная информационная система – это совокупность программно-аппаратных средств, предназначенных для автоматизации деятельности, связанной с хранением, передачей и обработкой информации.

Основной причиной создания СДО является:

Предоставление школьникам, студентам, гражданским и военным специалистам, безработным, самым широким кругам населения равных образовательных возможностей в любых районах страны и за ее рубежами;

Повышение качественного уровня образования за счет более активного использования научного и образовательного потенциала ведущих университетов, академий, институтов, лидирующих отраслевых центров подготовки и переподготовки кадров, институтов повышения квалификации, других образовательных учреждений;

Возможность получения как базового, так и дополнительного образования параллельно с основной деятельностью, расширение образовательной среды в России на наиболее полное удовлетворение потребностей и прав человека в области образования, интеграция с очной и заочной формами обучения, совершенствуя и развивая их, создание условий для непрерывного образования;

Обеспечение принципиально нового уровня доступности образования при сохранении его качества.

UML – язык графического описания для объектного моделирования в области разработки программного обеспечения для моделирования, бизнес-процессов, системного проектирования и отображения организационных структур.

UML позволяет разработчикам программного обеспечения достигнуть соглашения в графических обозначениях для представления общих понятий и больше сконцентрироваться на проектировании и архитектуре.

C# - Объектно-ориентированный язык программирования имеющий статическую типизацию и поддерживающий полиморфизм, перегрузку операторов, делегаты, атрибуты, события, свойства, обобщенные типы и методы и т.д. Переняв многое от своих предшественников – языков C++, Pascal, Smalltalk и Java – C#, опираясь на практику их использования, исключает некоторые модели, зарекомендовавшие себя как проблематичные при разработке программных систем, например отказ от поддержки множественного наследования классов.

* 1. **Анализ существующих аналогов**

1.2.1 [iSpring Online](https://lmslist.ru/sdo/obzor-ispring-online/)

«[iSpring Online](https://lmslist.ru/sdo/obzor-ispring-online/) » СДО iSpring Online проста в использовании и обладает интуитивно понятным интерфейсом, и отзывы пользователей это подтверждают. СДО удобно использовать совместно с редактором курсов [iSpring Suite](http://www.ispring.ru/ispring-suite), который позволяет создавать интерактивные курсы, тесты и диалоговые тренажеры и затем в три клика загружать их в iSpring Online. Оба решения имеют бесплатный 14-дневный ознакомительный период — начать обучать сотрудников можно сразу после регистрации аккаунта (рис. 1.1).

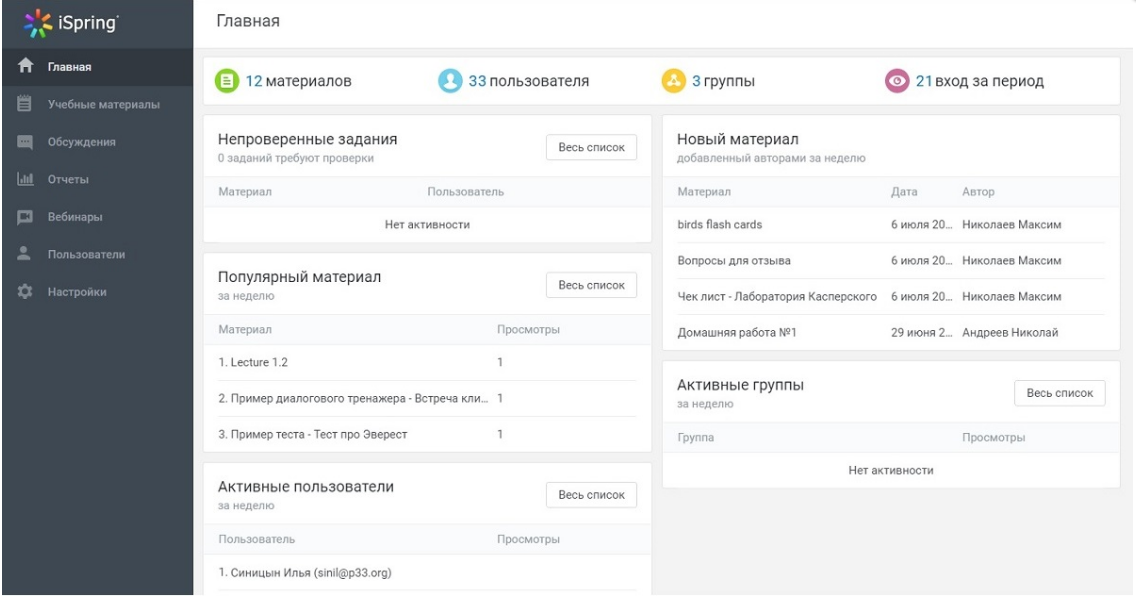


Рисунок 1.1 – [iSpring Online](https://lmslist.ru/sdo/obzor-ispring-online/)

1.2.2 [Moodle](https://moodle.org/?lang=ru)

[Moodle](https://moodle.org/?lang=ru) полностью бесплатен – его можно свободно скачивать, устанавливать, изменять и т.д. Он относиться к Open Source системам, т.е. системам с открытым исходным кодом, что позволяет многим программистам создавать дополнительные, очень полезные расширения или модули. Стоит отметить, что многие крупные вузы РФ используют Moodle в своей работе (рис. 1.2).

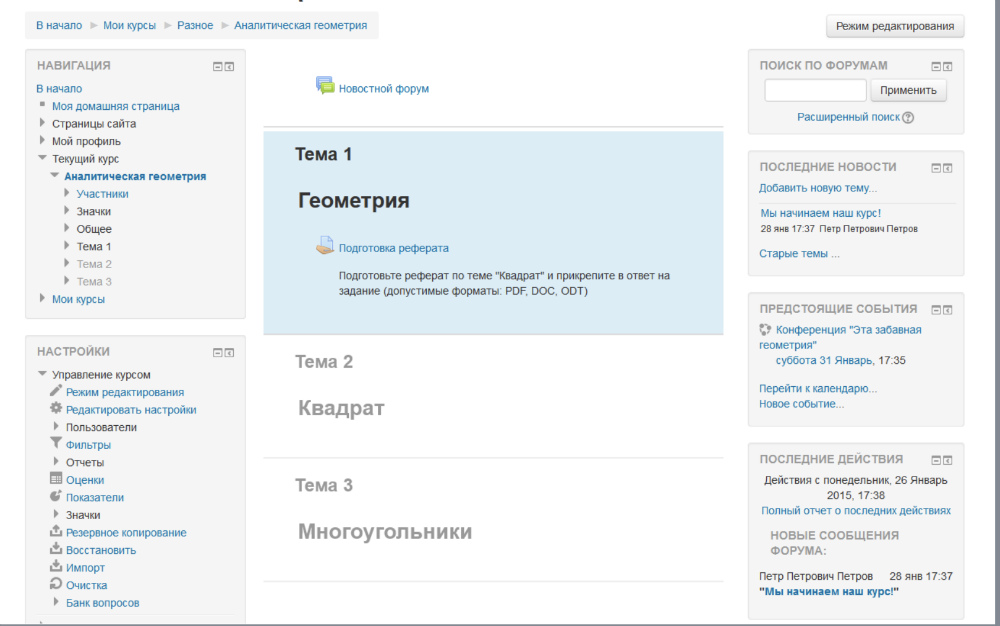


Рисунок 1.2 – Главная страница.

1.2.3 [ShareKnowledge](http://competentum.ru/sistema-distancionnogo-obucheniya)

открытым исходным кодом, что позволяет многим программистам создавать дополнительные, очень полезные расширения или модули. Стоит отметить, что многие крупные вузы [ShareKnowledge](http://competentum.ru/sistema-distancionnogo-obucheniya) – система дистанционного обучения от международной группы компаний Competentum.

Группа Competentum — эксперт в области разработки электронных учебных курсов. Компания «ФИЗИКОН», входящая в группу, разрабатывает курсы для школьников и студентов с 1994 года. На начало 2017 года в их библиотеке уже более 20 тысяч учебных материалов.

Группа Competentum также занимается разработкой решений для дистанционного обучения и автоматизации HR-процессов. Так в 2007 году в результате развития платформы Competentum.Instructor появилась СДО ShareKnowledge — первая в мире СДО, построенная на базе платформы Microsoft SharePoint (рис. 1.3)

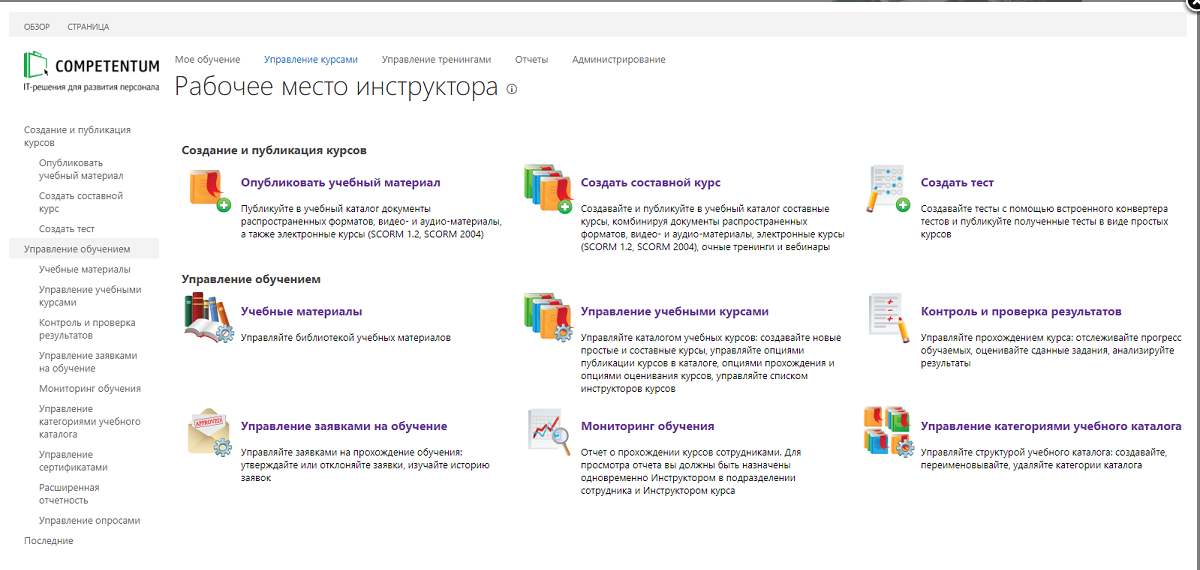


Рисунок 1.3 – Главное меню.

1.2.4 [TeachBase](https://lmslist.ru/sdo/obzor-teachbase/)

Облачный сервис [TeachBase](https://lmslist.ru/sdo/obzor-teachbase/) — разработка молодой московской компании «Интернет-школа»

Стартап TeachBase появился в 2012 году в результате постепенного развития проекта для онлайн-свиданий «Говорун», который сначала превратился в проект для подготовки школьников к ЕГЭ и затем — в b2b-платформу по обучению сотрудников компаний.

TeachBase — это простая в использовании система дистанционного обучения, которая обладает всеми необходимыми базовыми функциями: можно формировать курсы из готовых электронных учебных материалов, создавать тесты и распространять их среди пользователей системы, проводить видео встречи и анализировать процесс обучения (рис. 1.4).

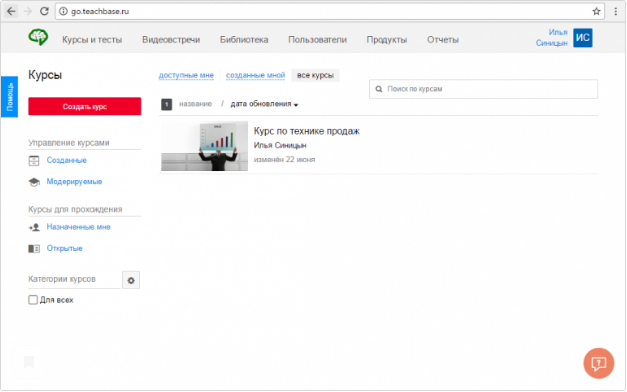


Рисунок 1.4 – Главное меню.

* 1. **Техническое задание**

Проанализировав приведенные выше аналоги, я пришел к выводу, что моей программе необходимы:

1. Функциональность – хранение, накопления и обновление данных, а так же возможность создания нового учебного пособия.
2. Безопасность – создать закрытую систему авторизации под контролем администрации.
3. Простота эксплуатации – реализовать простой для восприятия интерфейс.
4. **ТЕХНОЛОГИЧЕСКИЙ РАЗДЕЛ**
   1. **Назначение разработки**

Цель: Разработка системы дистанционного обучения с целью хранения, передачи, удаления и редактирования информации, а также создания тестевых и теоретических заданий.

Задачи:

1. Создать Базу Данных,

3. Подключить Базу Данных к форме,

4. Разработать администрирование,

5. Визуализировать программу.

* 1. **Требования к проекту:**
     1. Функциональные характеристики

Программа должна иметь возможность добавления, хранения, удаления и изменения содержимого. В качестве информационной сущности выступают учебники, пособия залитые в общий доступ. Ученики могут прочесть теоретический материал что был выдан им на уроке или повторить пройденное. Учителя смогут заливать весь теоретический материал в СДО и размещать домашние задания.

* + 1. Технические характеристики

СДО оптимизировано под любой браузер и не требует сторонних програм.

* + 1. Безопасность данных

Защита, будет осуществляется через авторизацию пользователя по коду, который будет служить паролем. Регистрацией новых пользователей, будет занимается непосредственно сис – отдел учебного учреждения.

* + 1. Графический интерфейс

Визуальная составляющая программы ещё будет примерно такой:

1. Окно авторизации (рис. 2.1),
2. Главная страница учителя (рис 2.2),
3. Главная страница ученика (рис 2.3).

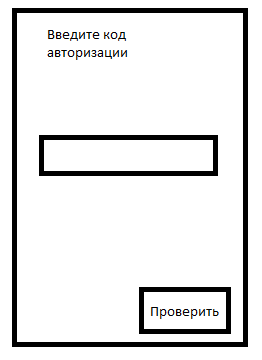


Рисунок 2.1 - Окно авторизации».

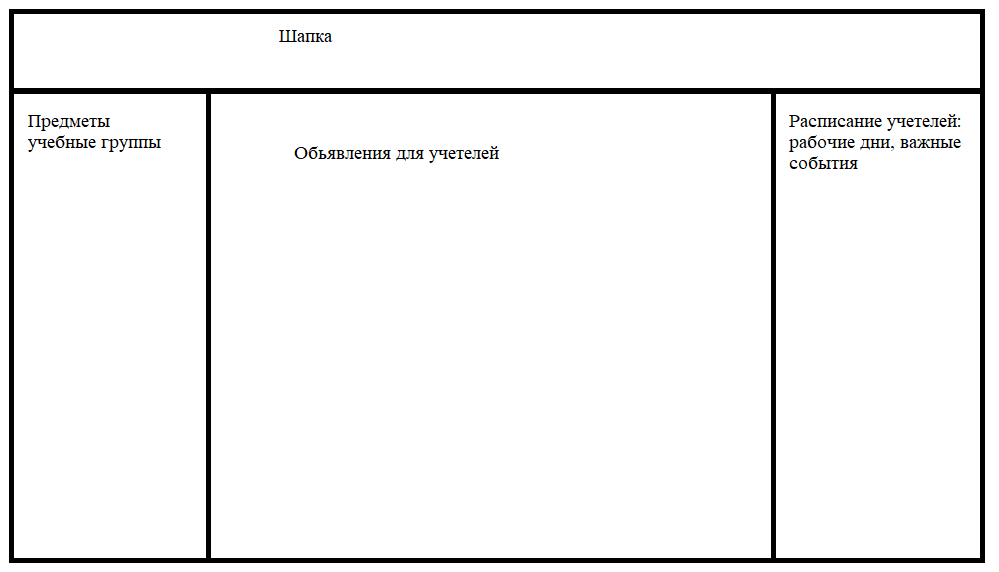


Рисунок 2.2 – Главная страница учителей.

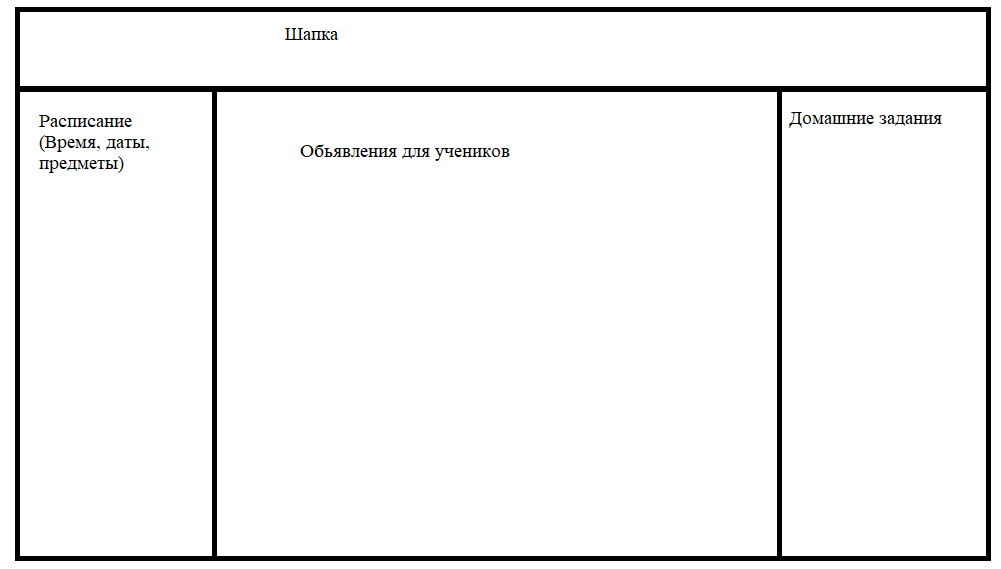


Рисунок 2.3 – Главная страница учеников.

* 1. **Выбор программных и технических средств**

В качестве среды разработки использовалась программа Visual Studio, которая включает в себя редактор исходного кода и возможность его рефакторинга. В программе можно создать форму приложения и его составляющие компоненты включая саму Базу Данных. Правда для БД необходимо поставить дополнительную функцию о которой позже пойдет речь.

Для создания Базы данных применялся Microsoft SQL Server.

MS SQL представляет собой систему управления реляционными базами данных использующую, в основном, язык запросов Transact-SQL. Используется для работы с базами данных размером от персональных до крупных баз данных масштаба предприятия

Выбор пал на конкретно эту СУРБД потому, что в ней присутствуют функции отказоустойчивости, а именно зеркалирование и наличие кластеров высокой доступности, масштабируемости – репликации, распределения ресурсов и безопасности – интерактивного шифрования баз данных.

* 1. **Проектирование системы**

Диаграмма прецедентов – диаграмма, отражающая отношения между актерами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне (рис. 2.4).

Прецедент – возможность моделируемой системы, благодаря которой пользователь может получить конкретный, измеримый и нужный ему результат.

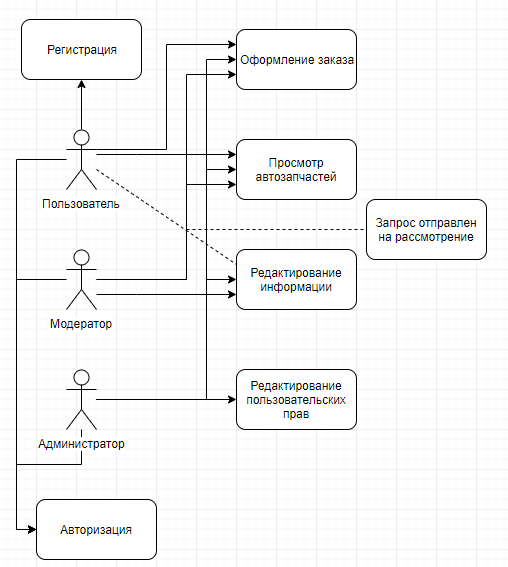


Рисунок 2.4 – Диаграмма прецедентов

Диаграмма классов – структурная диаграмма языка моделирования UML, демонстрирующая общую структуру иерархии классов системы, их коопераций, атрибутов, методов, интерфейсов и взаимосвязей между ними (рис. 2.5).

Целью создания диаграммы классов является графическое представление статической структуры декларативных элементов системы.

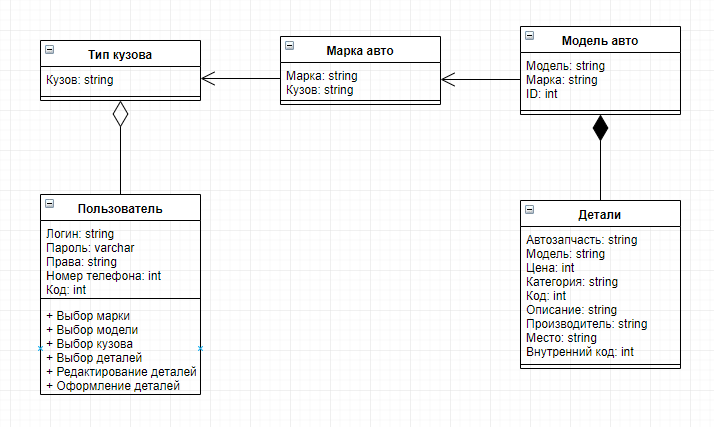


Рисунок 2.5 – Диаграмма классов.

Диаграмма последовательности – диаграмма, на которой для некоторого набора объектов на единой временной оси показан жизненный цикл какого-либо определенного объекта и взаимодействие актеров ИС в рамках какого-либо определенного прецедента (рис 2.6).

На данной диаграмме объекты располагаются слева направо.

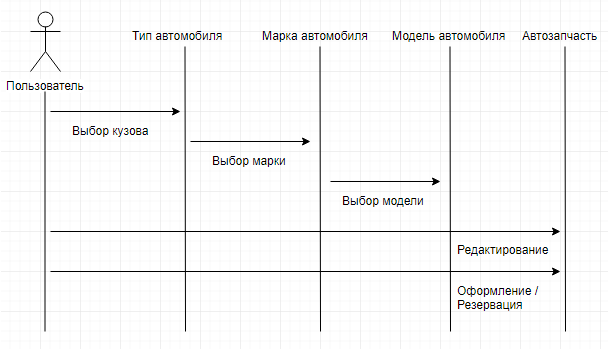


Рисунок 2.6 – Диаграмма последовательности.

ER модель – модель данных, позволяющая описывать концептуальные схемы предметной области. Используется при высокоуровневом проектировании баз данных. С её помощью можно выделить ключевые сущности и обозначить связи, которые могут устанавливаться между этими сущностями (рис 2.7).

Во время проектирования баз данных происходит преобразование ER-модели в конкретную схему базы данных на основе выбранной модели данных.

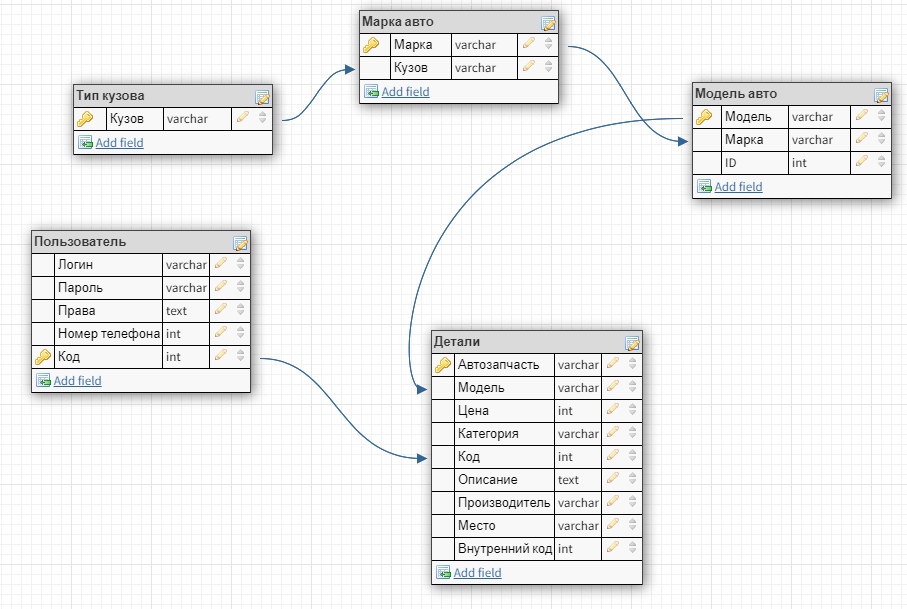


Рисунок 2.7 – ER модель.

Диаграмма развертывания – диаграмма, моделирующая физическое развертывание на узлах. Она показывает, какие аппаратные компоненты существуют, какие программные компоненты работают на каждом узле и как различные части этого комплекса соединяются друг с другом (рис. 2.8).

Узлы представляются как прямоугольные параллелепипеды с артефактами, расположенными в них, изображенными в виде прямоугольников.

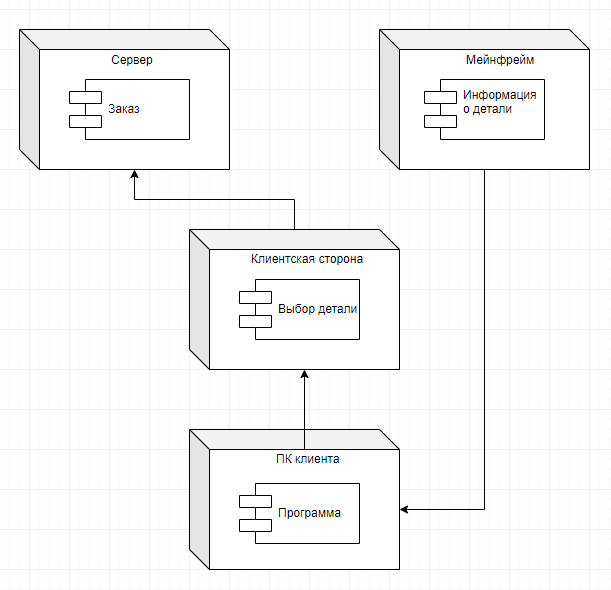


Рисунок 2.8 – Диаграмма развертывания.

1. **ОРГАНИЗАЦИОННЫЙ РАЗДЕЛ**
   1. **Организация графика разработки**

Trello — это одна из самых популярных систем управления проектами в режиме онлайн, которая пользуется особенным спросом среди небольших компаний и стартапов. Она позволяет эффективно организовывать работу по японской методологии канбан-досок.

Она создана Fog Creek Software в 2011 году на базе MongoDB, Backbone.js и Node.js. Главные достоинства, которые позволили Trello добиться популярности — это:

1. Простой интерфейс;
2. Почти неограниченный бесплатный доступ;
3. Удобство в работе и возможность интеграции с другими популярными инструментами для онлайн-работы.

Для организации задач используется доска с карточками, которые распределяются по типам. Как правило, задачи разбиваются на:

1. Запланированные,
2. Текущие,
3. Выполненные.

Это самая элементарная структура, возможности по модернизации которой ограничены лишь вашим воображением.

Далее пойдут скриншоты, включающие в себя начало работы (рис. 3.1), ситуацию в процессе (рис. 3.2), конец первой (рис. 3.3), второй (рис. 3.4) частей и финальный этап (рис. 3.5).

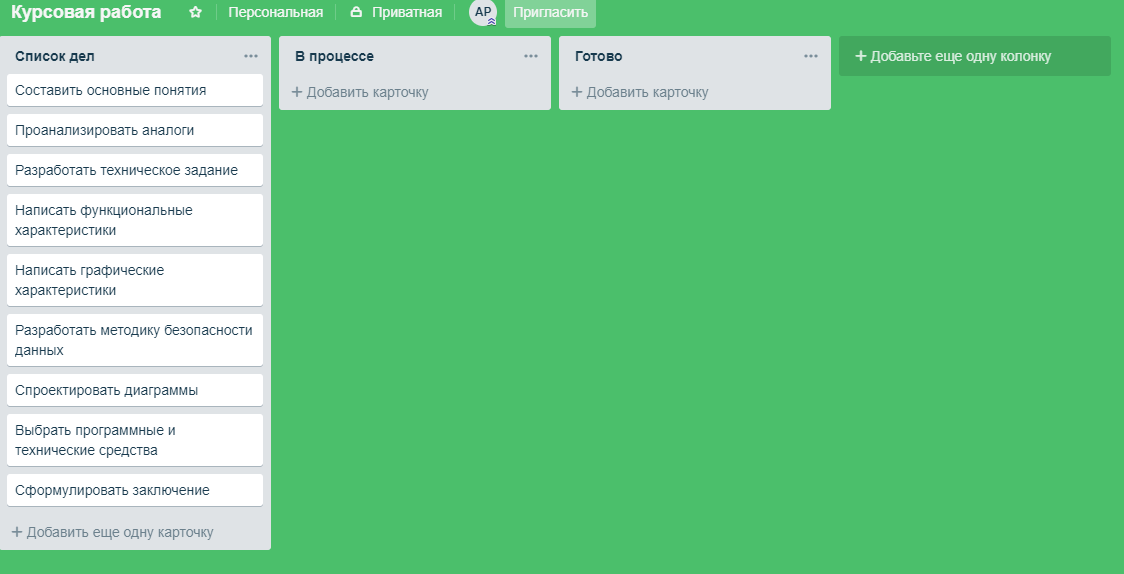


Рисунок 3.1 – Начало проектирования.

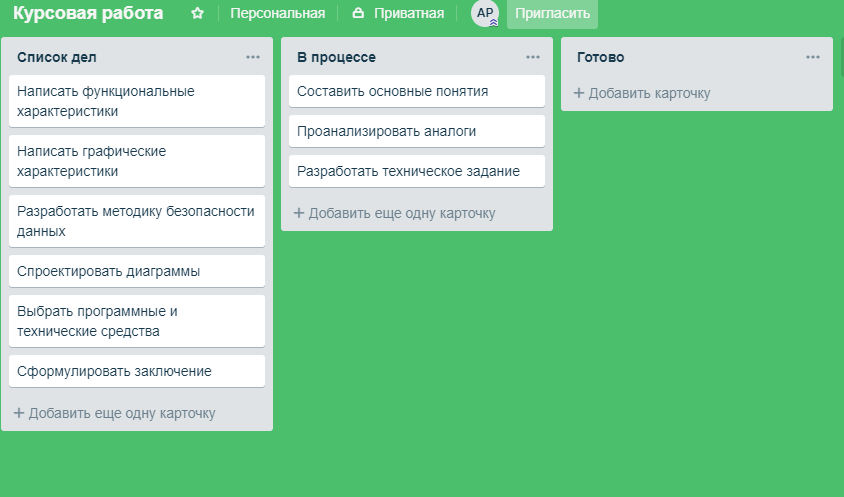


Рисунок 3.2 – В процессе.

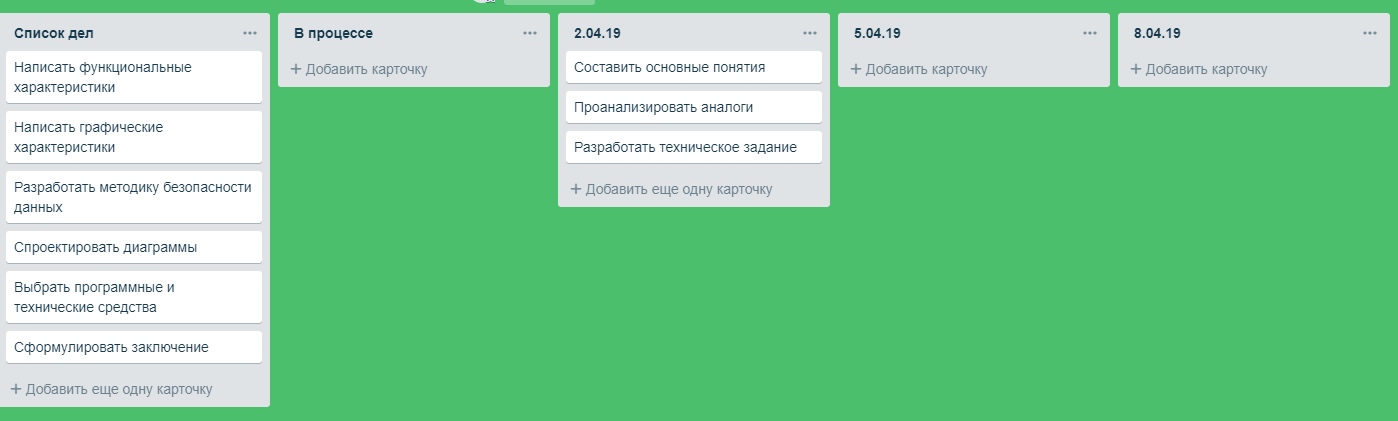


Рисунок 3.3 – Конец первой части.

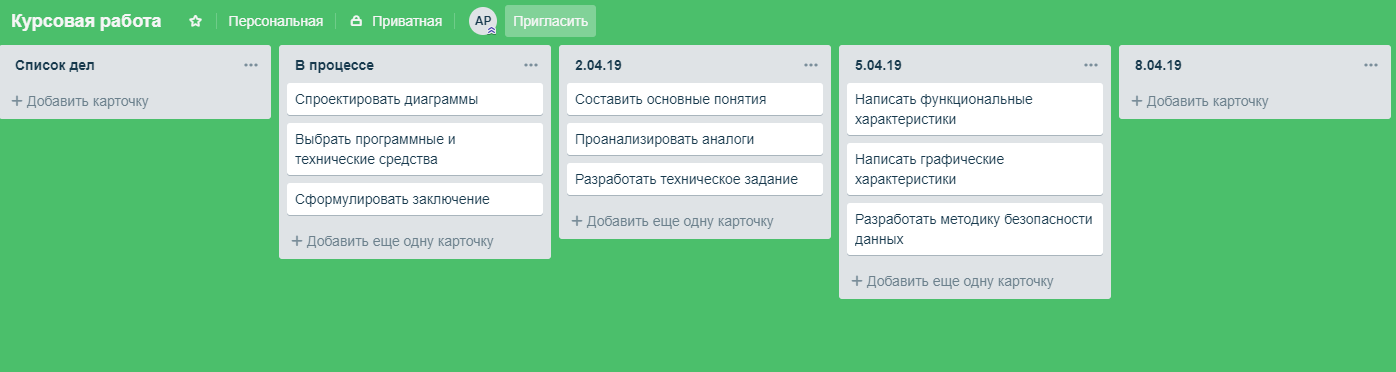


Рисунок 3.4 – Конец второй части.

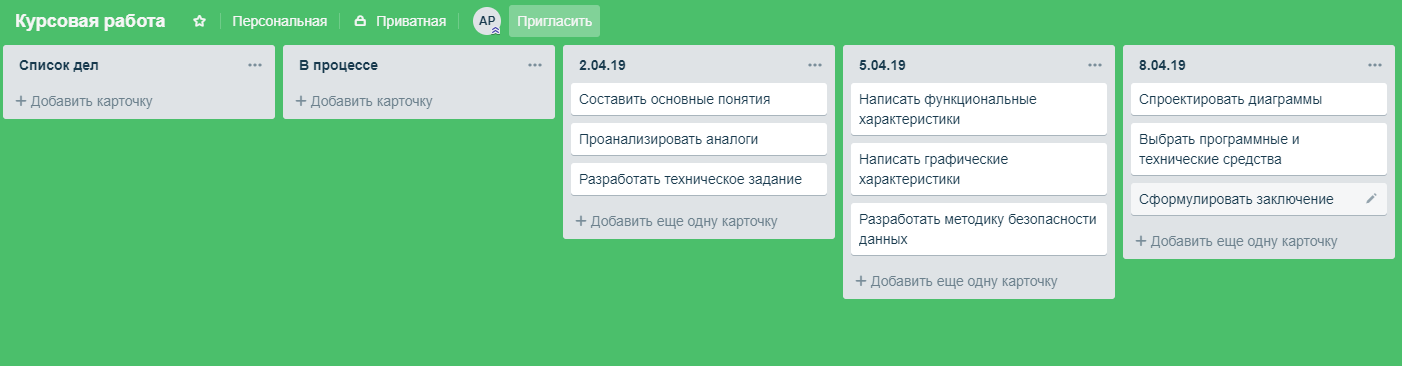


Рисунок 3.5 – Завершение работы.

* 1. **Система контроля версий**

Github – крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки.

Создатели сайта называют GitHub «[социальной сетью](https://ru.wikipedia.org/wiki/%D0%A1%D0%BE%D1%86%D0%B8%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D1%81%D0%B5%D1%82%D1%8C) для разработчиков». Кроме размещения кода, участники могут общаться, комментировать правки друг друга, а также следить за новостями знакомых. С помощью широких возможностей Git программисты могут объединять свои [репозитории](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BF%D0%BE%D0%B7%D0%B8%D1%82%D0%BE%D1%80%D0%B8%D0%B9" \o "Репозиторий) — GitHub предлагает удобный интерфейс для этого и может отображать вклад каждого участника в виде [дерева](https://ru.wikipedia.org/wiki/%D0%94%D0%B5%D1%80%D0%B5%D0%B2%D0%BE_(%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D1%8F_%D0%B3%D1%80%D0%B0%D1%84%D0%BE%D0%B2)).

Для проектов есть личные страницы, небольшие [Вики](https://ru.wikipedia.org/wiki/%D0%92%D0%B8%D0%BA%D0%B8) и [система отслеживания ошибок](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D0%BE%D1%82%D1%81%D0%BB%D0%B5%D0%B6%D0%B8%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F_%D0%BE%D1%88%D0%B8%D0%B1%D0%BE%D0%BA). Прямо на сайте можно просмотреть файлы проектов с [подсветкой синтаксиса](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D0%B4%D1%81%D0%B2%D0%B5%D1%82%D0%BA%D0%B0_%D1%81%D0%B8%D0%BD%D1%82%D0%B0%D0%BA%D1%81%D0%B8%D1%81%D0%B0) для большинства [языков программирования](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F).

1. На платных тарифных планах можно создавать приватные репозитории, доступные ограниченному кругу пользователей.
2. Есть возможность прямого добавления новых файлов в свой репозиторий через веб-интерфейс сервиса.
3. Код проектов можно не только скопировать через [Git](https://ru.wikipedia.org/wiki/Git" \o "Git), но и скачать в виде обычных архивов с сайта.
4. Кроме [Git](https://ru.wikipedia.org/wiki/Git" \o "Git), сервис поддерживает получение и редактирование кода через [SVN](https://ru.wikipedia.org/wiki/SVN) и [Mercurial](https://ru.wikipedia.org/wiki/Mercurial" \o "Mercurial).
5. На сайте есть [pastebin](https://ru.wikipedia.org/wiki/Pastebin)-сервис [gist.github.com](https://gist.github.com/) для быстрой публикации фрагментов кода.

Ранее [Ruby](https://ru.wikipedia.org/wiki/Ruby)-проекты могли быть автоматически опубликованы в [RubyGems](https://ru.wikipedia.org/wiki/RubyGems" \o "RubyGems)-репозитории сервиса, но в октябре 2009 GitHub отказался от этого сервиса.

Также GitHub может похвастаться контролем доступа, багтрекингом, управлением задачами и вики для каждого проекта. Цель GitHub — содействовать взаимодействию разработчиков.

К проекту, загруженному на GitHub, можно получить доступ с помощью интерфейса командной строки Git и Git-команд. Также есть и другие функции, такие как документация, запросы на принятие изменений (pull requests), история коммитов, интеграция со множеством популярных сервисов, email-уведомления, эмодзи, графики, вложенные списки задач, система упоминаний, похожая на ту, что в Twitter, и т.д.

На представленном ниже скриншоте видно, как будет выглядеть репозиторий, если в него добавить какие-либо файлы (рис. 3.6).

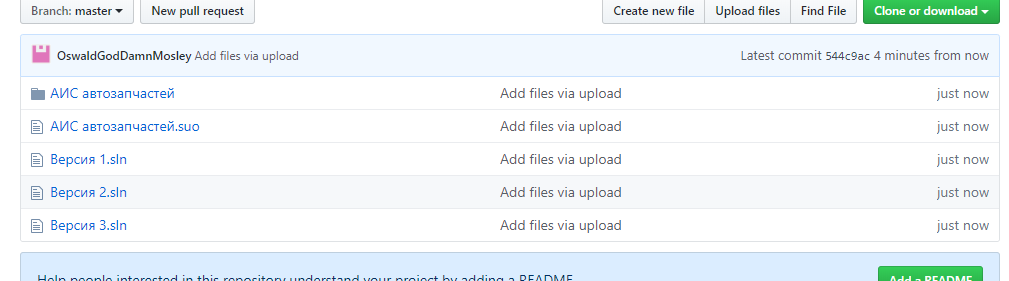


Рисунок 3.6 – Репозиторий.

**ЗАКЛЮЧЕНИЕ**

Чтобы достигнуть разработки АИС автозапчастей необходимо было выполнить:

1. Создать Базу Данных,

2. Спроектировать меню автомобилей,

3. Подключить Базу Данных к форме,

4. Разработать администрирование,

5. Визуализировать программу.

В процессе работы над проектом были получены практические навыки в исследовании предметной области, описания проектного решения, построения моделей.

Выполнение задач оценивается положительно, хотя стоит отметить что это ещё не конец и программа будет совершенствоваться со временем.

**СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ**

1. ГОСТ 2.004-88 ЕСКД. Общие требования к выполнению конструкторских и технологических документов на печатающих устройствах вывода ЭВМ;
2. Сайт Stab 24 [Электронный ресурс] / Режим доступа:

http://stud24.ru/economics/analiz-sistem-distancionnogo-obucheniya-v/158876-464521-page1.html Дата обращения – (11.04.2019).

1. Сайт Lmslist [Электронный ресурс] / Режим доступа:

https://lmslist.ru/sdo/ Дата обращения – (11.04.2019)

1. Сайт StudFiles [Электронный ресурс] / Режим доступа:

https://studfiles.net/preview/274954/page:3/ – (11.04.2019)