Министерство цифрового развития

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Сибирский государственный университет телекоммуникаций и информатики»

(СибГУТИ)

Кафедра математического моделирования и цифрового развития бизнес‑систем

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту по дисциплине «Проектный практикум»

на тему: «Проектирование ИС по подбору образовательного контента»

Исполнитель:

студент гр. ИИ-862 \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Юндунов К.А.

(подпись) ( ФИО )

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_ доц. каф. ММиЦРБС Полетайкин А.Н.

(подпись) ( должность, ФИО )

Оценка \_\_\_\_\_\_\_ Дата защиты \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Новосибирск

2021

Министерство цифрового развития

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Сибирский государственный университет телекоммуникаций и информатики»

(СибГУТИ)

Кафедра математического моделирования и цифрового развития бизнес‑систем

ЗАДАНИЕ НА КУРСОВОЙ ПРОЕКТ

по дисциплине “Проектный практикум”

студенту Юндунову Киму Александровичу группы ИИ-862

1. Тема курсового проекта: Проектирование ИС по подбору образовательного контента

2. Перечень исходных материалов для курсового проектирования: специализированная литература, материалы сети Интернет

3. Используемое средство разработки приложения: VS Code

4. Основные функции приложения: сбор данных, обработка данных, структурирование данных на сервере

5. Тип используемой СУБД: Реляционная

6. Основные сущности БД: Данные о студентах, преподавателях и видеокурсах

7. Основные результаты работы приложения: Структуризация собранных данных о курсах и видеокурсах

Срок сдачи проекта на проверку и защиты: с 20.12.2021 по 26.12.2021 г.

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ доц. каф. ММиЦРБС Полетайкин А.Н.

**РЕФЕРАТ**

Курсовой проект: 52 стр., 39 рис., 8 табл., 4 прил., 5 ист.

В рамках реализации новой стратегии цифровой трансформации науки и высшего образования предусматривается переход к индивидуальной образовательной траектории (ИОТ) всех высших учебных заведений (ВУЗов) к 2030 году. Для построения ИОТ нужны данные об абитуриенте – результаты психологических тестирований, сведения о внеучебной и учебной (в случае, если имеется образование выше среднего общего) деятельности.

Объектом информатизации является приемная комиссия ВУЗа, а процессом информатизации – процесс сбора и обработки данных о внеучебной деятельности абитуриента.

Цель работы: создание системы для сбора и сохранения структурированных данных о внеучебной деятельности абитуриента.

Главным средством разработки выступает программная среда Open Server Panel.

Результатом работы является приобретение навыков web-разработки, а также функционирующее web-приложение, выполняющее функции сбора и сохранения данных, размещенное на локальном сервере.

Ключевые слова: абитуриент, внеучебная деятельность, web-приложение, индивидуальная образовательная траектория.

**Оглавление**

[ВВЕДЕНИЕ 5](#_Toc92969271)

[1. Характеристика объекта информатизации 6](#_Toc92969272)

[2. Анализ существующих компьютерных разработок 7](#_Toc92969273)

[3. Техническое задание на создание ИС 10](#_Toc92969274)

[4. Разработка функциональной структуры ИС 13](#_Toc92969275)

[5. Разработка информационного обеспечения ИС 19](#_Toc92969276)

[5.1 Обосновать выбор средства управления данными. 19](#_Toc92969277)

[5.2 Обосновать выбор средства разработки логической модели данных. 20](#_Toc92969278)

[5.3 Выполнить разработку логической модели данных. 20](#_Toc92969279)

[5.4 Описать разработку физической модели данных. 21](#_Toc92969280)

[5.5 Описать процесс организации сбора и обработки информации. 23](#_Toc92969281)

[6. Разработка математического обеспечения ИС 23](#_Toc92969282)

[7. Разработка программного обеспечения ИС 25](#_Toc92969283)

[7.1 Провести обоснованный выбор операционной системы для сервера и рабочих станций. 25](#_Toc92969284)

[7.2 Провести обоснованный выбор средства разработки специального ПО. 26](#_Toc92969285)

[7.3 Выбрать для обслуживания ИС сервисные программы и утилиты. 26](#_Toc92969286)

[7.4 Разработать структуру специального программного обеспечения. 26](#_Toc92969287)

[7.5 Составить и коротко описать схему взаимодействия БД и специального ПО. 28](#_Toc92969288)

[8. Разработка технического обеспечения ИС 28](#_Toc92969289)

[8.1 Разработать решения по выбору конфигурации компьютеров для сервера и рабочих станций. 28](#_Toc92969290)

[8.2 Выбрать топологию компьютерной сети. 29](#_Toc92969291)

[8.3 Выполнить выбор прочих технических устройств, задействованных в ИС. 32](#_Toc92969292)

[8.4 Разработать структурную схему комплекса технических средств (КТС). 32](#_Toc92969293)

# ВВЕДЕНИЕ

В связи с обостренной ситуацией в мире, связанный с короновирусной инфекцией, многие предприятия, компании и учебные заведения переходят на удаленный формат обучения и введения бизнеса. Альтернативой очного обучения стало введение стримов, обучающих видеороликов в интернете.

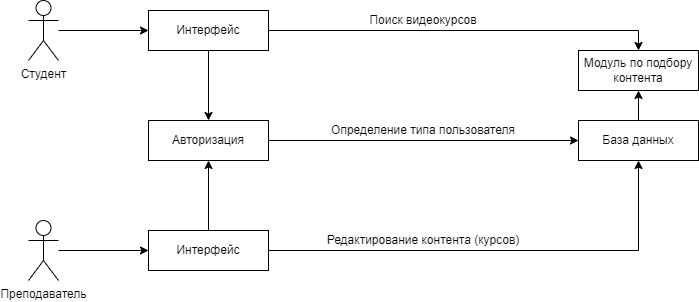
Видеохостинг — веб-сайт, позволяющий загружать и просматривать видео в браузере, например через специальный плеер. При этом большинство подобных сервисов не предоставляют видео, следуя таким образом принципу user generated content. Видеохостинг стал набирать популярность вместе с распространением широкополосного доступа в интернет и развитием (и удешевлением) жёстких дисков.

Сервис Youtube, принадлежащий сегодня компании Google, был первым видеохостингом и совершил революцию — новым увлечением активных пользователей Интернета стал просмотр видеосюжетов онлайн.

В данной работе представлена информационная система подбора видеоконтента для обучения в дистанционном формате, предназначенная для студентов и преподавателей.

# 1. Характеристика объекта информатизации

Данная система может не обслуживаться, а лишь дополняться контентом. Преподаватель обновляет контент (видеокурсы) по мере необходимости и составляет свой курс. Студент может лишь просмотреть данный контент по курсу. Каждый пользователь может авторизоваться и указать тип юзера (студент, преподаватель) для определения функционала. Для каждого студента, в зависимости от поведения (просмотры курсов), будет выводиться подборка видеокурсов.

****

Пользователь (студент/преподаватель), авторизуется для определения типа юзера.

Студент выполняет поиск для просмотра видеокурсов с помощью веб интерфейса.

Преподаватель загружает видеокурсы с помощью веб интерфейса.

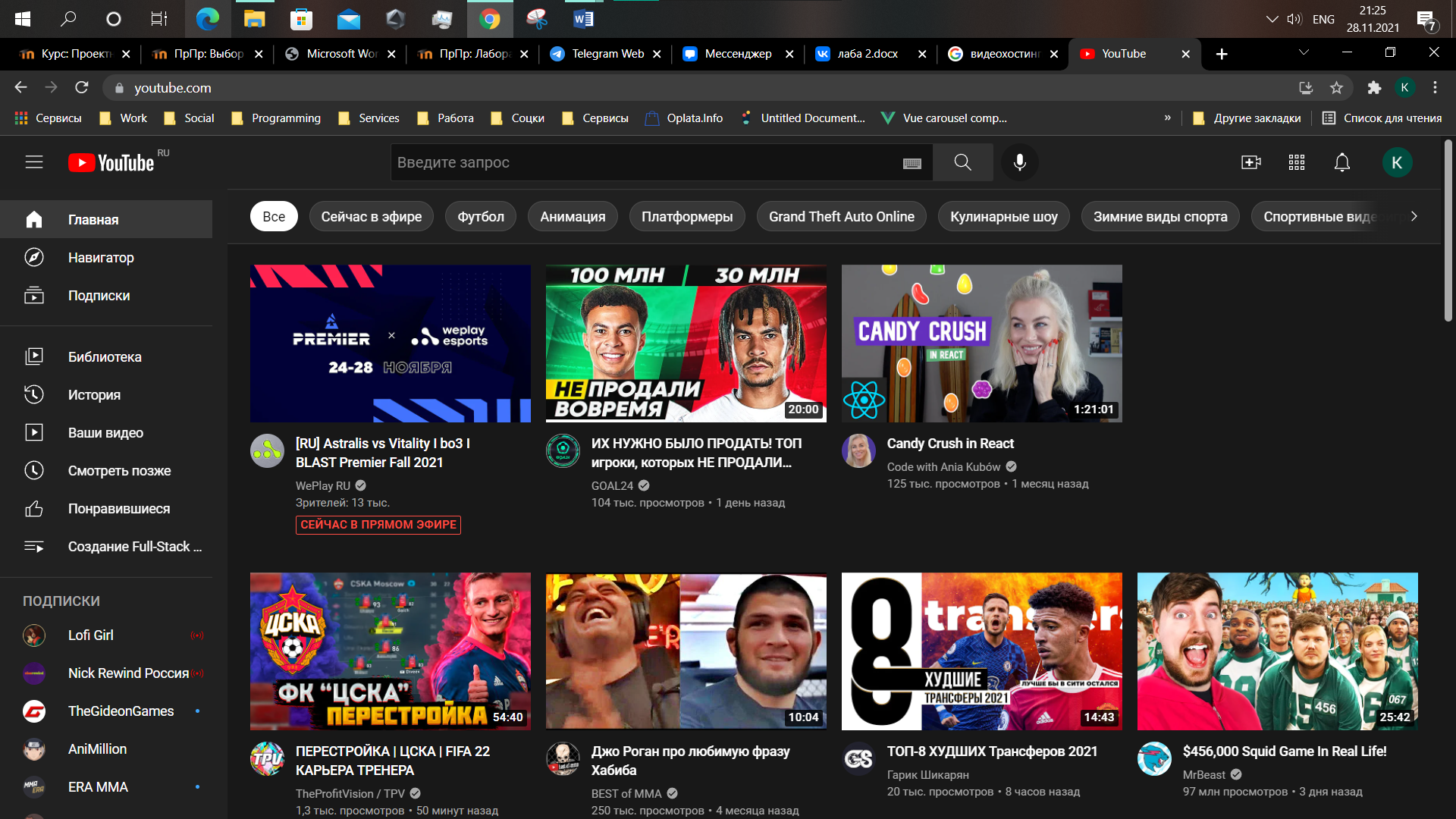
Аналогом данной ИС является упомянутый выше youtube, но наша система будет более узконаправленной.

Данную ИС можно усовершенствовать с помощью искусственного интеллекта в модуле подбора контента.

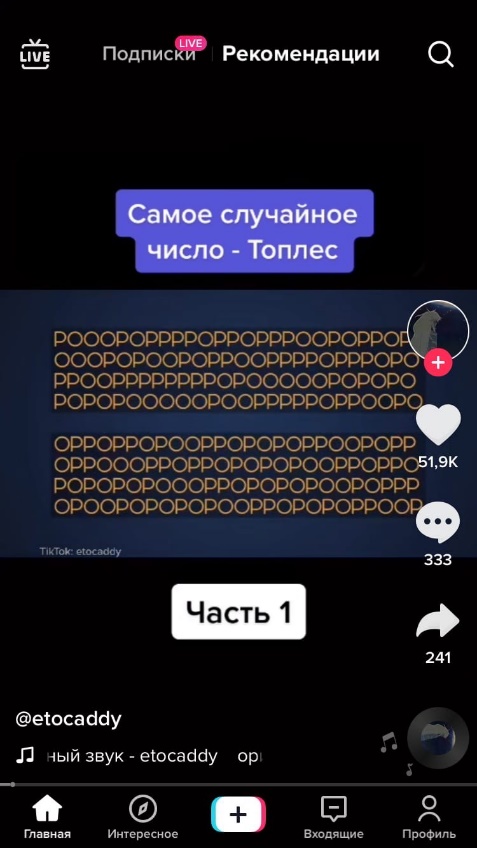
# 2. Анализ существующих компьютерных разработок

Аналоги данной ИС

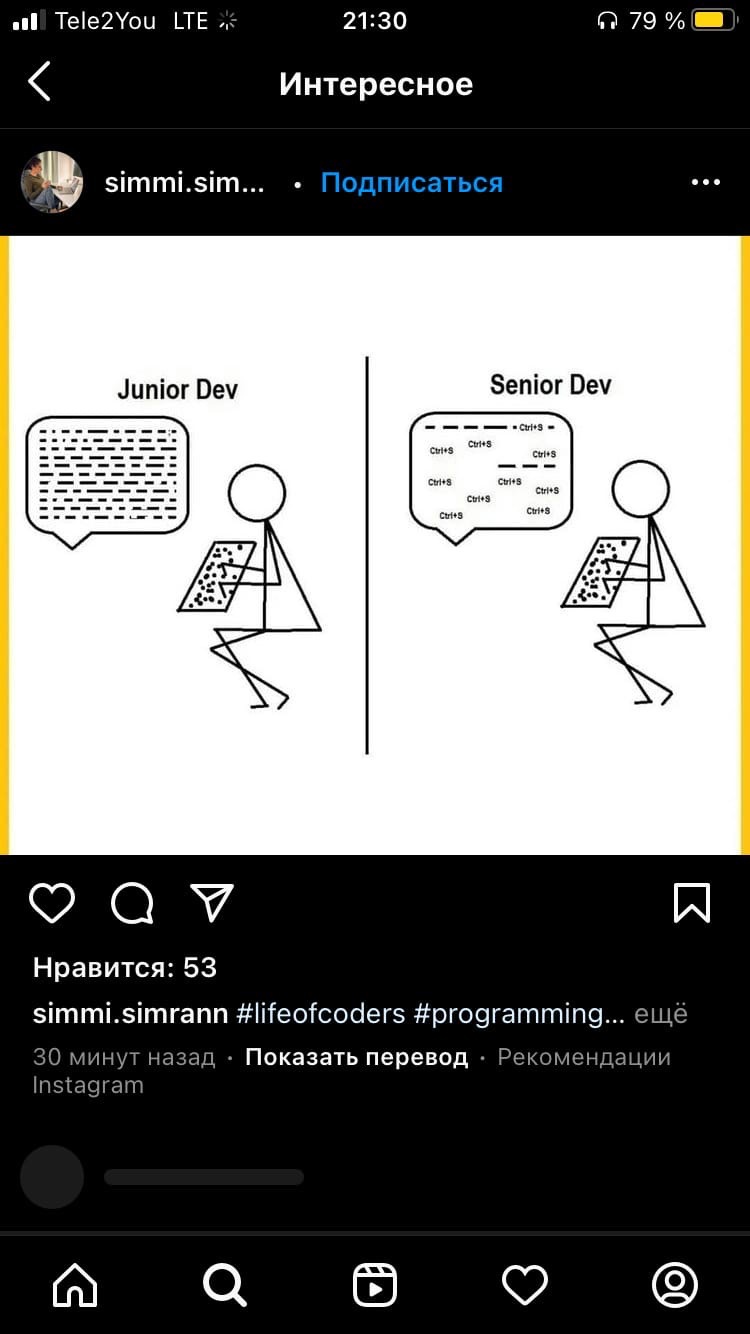
Видеохостинг Youtube. Самый популярный видеохостинг, но направленный на разные тематики, в нашей же, будет только образовательный контент.



Tiktok – ИС с короткими видео (обычно не более минуты), на данный момент самое скачиваемое приложение в мире.



Instagram – ИС с фото/видео контентом, долгое время держал топ-1 по скачиваниям, до прихода tiktok’а.



Рассматриваемые аналоги являются не узконаправленными, из-за чего данные системы можно не считать конкурентами нашей ИС, также к минусам данных ИС можно относить невозможность создания полноценного курса со своим описанием, названием и просмотрам статистики.

Сравним альтернативные ИС с нашим:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Критерий | наша ИС | Youtube | Tiktok | Instagram |
| Назначение ИС | 10/10 – т.к. узконаправленная ИС | 8/10 – можно, но много не нужного маетриала | 5/10 – короткие информативные ролики весьма удобны для студентов | 5/10 – т.к. и tiktok |
| эффективность системы | 10/10 | 8/10 | 5/10 | 5/10 |
| Гибкость системы | 10/10 | 9/10 – можно записывать и короткие и длинные видеокурсы | 1/10 – только короткие видеокурсы | 1/10 – только короткие видеокурсы |
| защищенность системы | 5/10, только хешированный пароль | 10/10 почта парль | 10/10 почта парль | 10/10 почта парль |
| живучесть системы | 5/10 | 10/10 – зарекомендовали себя на рынке | 10/10 – зарекомендовали себя на рынке | 10/10 – зарекомендовали себя на рынке |
| удобство пользовательского интерфейса системы | 5/10 – нет готового дизайна | 10/10 – есть веб версия, приложение на смратфоны | 10/10 – есть веб версия, приложение на смратфоны | 10/10 – есть веб версия, приложение на смратфоны |

# 3. Техническое задание на создание ИС

**Цель создания ИС**

ИС разработана для:

1. Полноценного перехода на формат дистанционного обучения
2. Удобного поиска и просмотра видеокурсов студентом
3. Создания курсов преподавателем

**Структура ИС и состав функциональных задач**



Состав функциональных задач:

1. Авторизация пользователя
   1. Требование в корректном JSON формате данных
   2. Требование к созданию сессий
   3. Требование к ui/ux отзывчивости
2. Поиск видеокурсов
   1. Требование к корректному заполнению поля поиска
   2. Требование к ui/ux отзывчивости
3. Создание видеокурсов
   1. Требование в корректном JSON формате данных
   2. Требование к ui/ux отзывчивости
4. Просмотр видеокурсов
   1. Требование к ui/ux отзывчивости
5. Редактирование видеокурсов
   1. Требование к ui/ux отзывчивости
6. Удаление видеокурсов
   1. Требование к ui/ux отзывчивости
7. Поиск курсов
   1. Требование к ui/ux отзывчивости
   2. Требование к корректному заполнению поля поиска
8. Создание курсов
   1. Требование к ui/ux отзывчивости
   2. Требование в корректном JSON формате данных
9. Просмотр курсов
   1. Требование к ui/ux отзывчивости
10. Редактирование курсов
    1. Требование к ui/ux отзывчивости
11. Удаление курсов
    1. Требование к ui/ux отзывчивости

**Требования к обеспечению ИС**

Требования к входной и выходной информации:

1. Использование кириллицы, латиницы, цифр и знаков препинания.
2. Ввод корректных данных
3. Вывод корректных данных

Требования к информационному обеспечению:

1. Вывод информации в формате JSON, html/css/js для браузера
2. Вывод запросов по REST API

Требования к математическому обеспечению:

1. Алгоритмы сортировки
2. Алгоритмы поиска

Требования к программному обеспечению:

1. Работа с базой данных через phpMyAdmin
2. Написание кода через vs code
3. Разработка ПО на платформе nodejs

Требования к техническому обеспечению:

1. Любой браузер поддерживаемый платформой (android, ios, windows, linux)

**Общие требования к подсистеме в целом**

Требования к надежности:

1. Vps/Vds сервера будут приобретены у надежных партнеров
2. При падении сервера, происходит бэкап (все данные будут сохранены)

Требования к безопасности:

1. Каждый отдельный юзер будет иметь хэшированный пароль
2. Каждый юзер будет проходить авторизацию/регистрацию для получения хэшированного пароля

Требования к численности и квалификации персонала и режиму его

Работы:

1. Нужен лишь один сисадмин для проверки состояния серверов

Требования к эксплуатации, техническому обслуживанию и хранению

компонентов подсистемы:

1. Т.к. мы покупаем облачные сервера, на сторонних сервисах, то требований нет.

Требования к защите информации от несанкционированного доступа:

1. Защита БД паролем
2. Защита юзера паролем и токеном сессии
3. Защита аккаунта для покупки облачных сервисов, паролем и почтой
4. Защита почты с помощью номера телефона и пароля
5. Защита номера телефона с помощью пароля на телефоне
6. Защита пароля на телефоне с помощью touch id (отпечатка пальца)

Требования по сохранению информации при авариях:

1. При аварии происходит бэкап сервера и данные сохраняются в облаке

# 4. Разработка функциональной структуры ИС

**Обоснование разрабатываемой ИС**

Источниками данных для ИС должны быть данные о пользователе, его тип (студент/преподаватель), единообразная БД для контента.

Система ведения и анализа результатов моделирования прочностных испытаний конструкций должна обеспечивать возможность выполнения следующих функций:

1. Ввод и вывод данных о пользователе (ЛК)
2. Ввод и вывод курсов для пользователя
3. Ввод и вывод видеокурсов для пользователя
4. Ввод и вывод статистики просмотров видеокурсов для преподавателей
5. Запись и редактирование курсов для преподавателей
6. Запись и редактирование видеокурсов для преподавателей

Разрабатываемая ИС должна иметь разграничение доступа. Система авторизации будет заключаться во вводе пользователем уникального имени, а также пароля, подтверждающего личность пользователя.

**Структурная организация**

В организационном аспекте управление проектированием рассматриваются по уровням организационно-административной структуры с соответствующими правами и обязанностями субъектов процесса проектирования.

Организация работ по проектированию ИС определяется порядком взаимодействия между несколькими сторонами, участвующими в этом процессе: пользователем, заказчиком, администратором и разработчиком.

Пользователь – это организация или группа подразделений, которые используют результаты обработки информации на ЭВМ.

Администратор – ответственное лицо, которое выполняет эксплуатацию программно-технических средств и информационного и методологического обеспечения ИС (технологические и инструкционные карты). Администратор несет ответственность перед пользователем за правильность результатов работы ИС и их своевременность, а перед заказчиком и разработчиком – за соблюдением условий эксплуатации, требований к технической документации.

Разработчик – это ответственное лицо (организация или подразделение), которое выполняет следующие функции:

разрабатывает ИС по техническому заданию заказчика;

принимает участие во внедрении;

осуществляет сдачу проекта заказчику;

осуществляет авторское сопровождение проекта.

Разработчик несет ответственность перед заказчиком за правильность реализации требований ТЗ на ИС, научно-технический уровень разработки, сроки проведения работ, качество проектной документации, правильность расхода денежных ресурсов.

**Описание процесса функционирования объекта в условиях**

**ИС**

Для функциональных задач: Авторизация пользователя, Поиск видеокурсов, Создание видеокурсов, Просмотр видеокурсов, Редактирование видеокурсов, Удаление видеокурсов, Поиск курсов, Создание курсов, Просмотр курсов, Редактирование курсов, Удаление курсов, общее описание:

1. Создание задачи в трекере (доска с задачами)
2. Составление архитектуры авторизации
3. Написание API для бэка
4. Создание дизайна для авторизации(форма регистрации/авторизации)
5. Верстка форм
6. Интеграция API с версткой
7. Деплой приложения на облачный сервер
8. Тестирование приложения
9. Запись выполненной задачи в доске

Диаграмма деятельности:

Диаграммы сделаны на в visual code т.к. на данный момент у меня стоит linux (ubuntu)

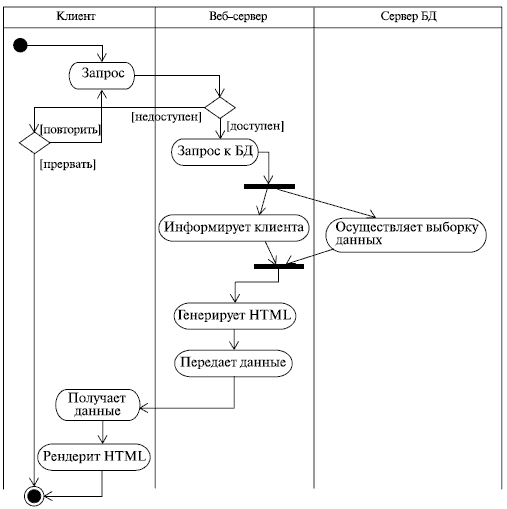
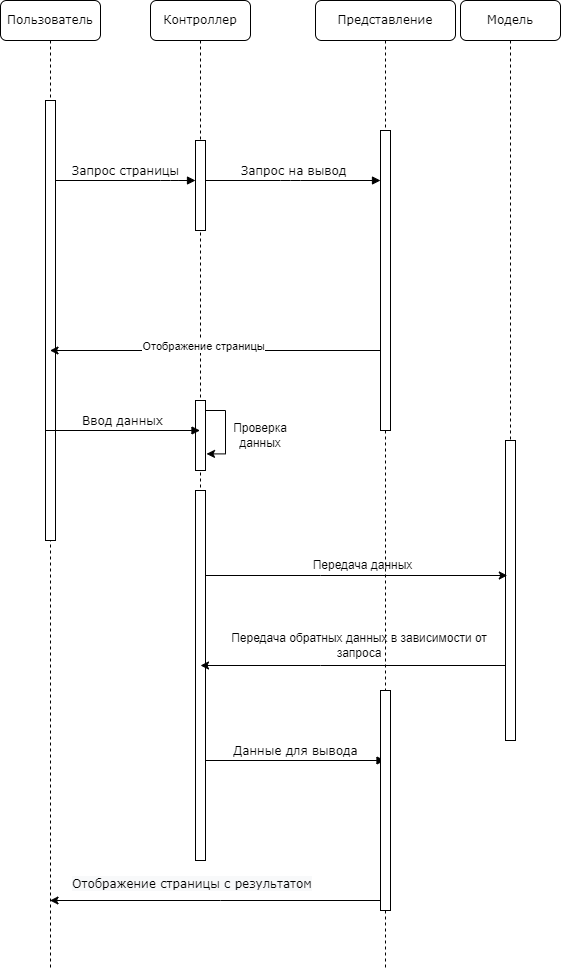


Диаграмма последовательности (по архитектуре MVC):

****

**Автоматизированные функции, исполняемые подсистемой**

Основными автоматизированными функциями нашей ИС являются:

**Просмотр видеокурсов**

Входные данные:

1. ID видеокурса, для поиска в БД курса по уникальному ключу

Выходные данные:

1. Видеокурс в видео формате
2. Название видеокурса
3. Описание видеокурса
4. Создатель видеокурса

**Создание/Редактирование видеокурсов**

Входные данные:

1. ID (для редактирования)
2. Видео
3. Название
4. Описание

Выходные данные:

1. Результат выполнения функции (200 статус http запроса)

**Просмотр курса:**

Входные данные:

1. ID курса

Выходные данные:

1. Название курса
2. Описание курса
3. Список видеокурсов по данному курсу

**Создание/Редактирование курса**

Входные данные:

1. ID курса (для редактирования)
2. Список видеокурсов по данному курсу
3. Список создателей курса
4. Описание курса
5. Название курса

Выходные данные:

1. Результат выполнения функции (200 статус http запроса)

**Личный кабинет**

Входные данные:

1. ID пользователя

Выходные данные:

1. Имя пользователя
2. Тип пользователя
3. Список курсов (для преподавателя)
4. Список видеокурсов (для преподавателя)
5. Список просмотренных видеокурсов

# 5. Разработка информационного обеспечения ИС

# Обосновать выбор средства управления данными.

Для того чтобы сделать выбор средства управления данными, сравним две системы: MySQL и MongoDB.

Плюсы MySQL:

1. Реляционная структура требует большего планирования и контроля
2. Данные легко использовать из разных приложений
3. Много приложений +15 лет
4. Гибкость
5. Огромное кол-во решений и простота использования + интегрирования

Плюсы MongoDB

1. Скорость разработки
2. Не нужно синхронизировать схему в базе данных и приложений
3. Понятный путь к масштабируемости
4. Простые предписанные решения

Для нашей ИС мы будем использовать MySQL так, как данная система управления данными бесплатна, а также используется для проектов, написанных с использованием языка программирования JavaScript. Для использования данной системы управления данными будет использоваться веб-интерфейс phpMyAdmin для MySQL.

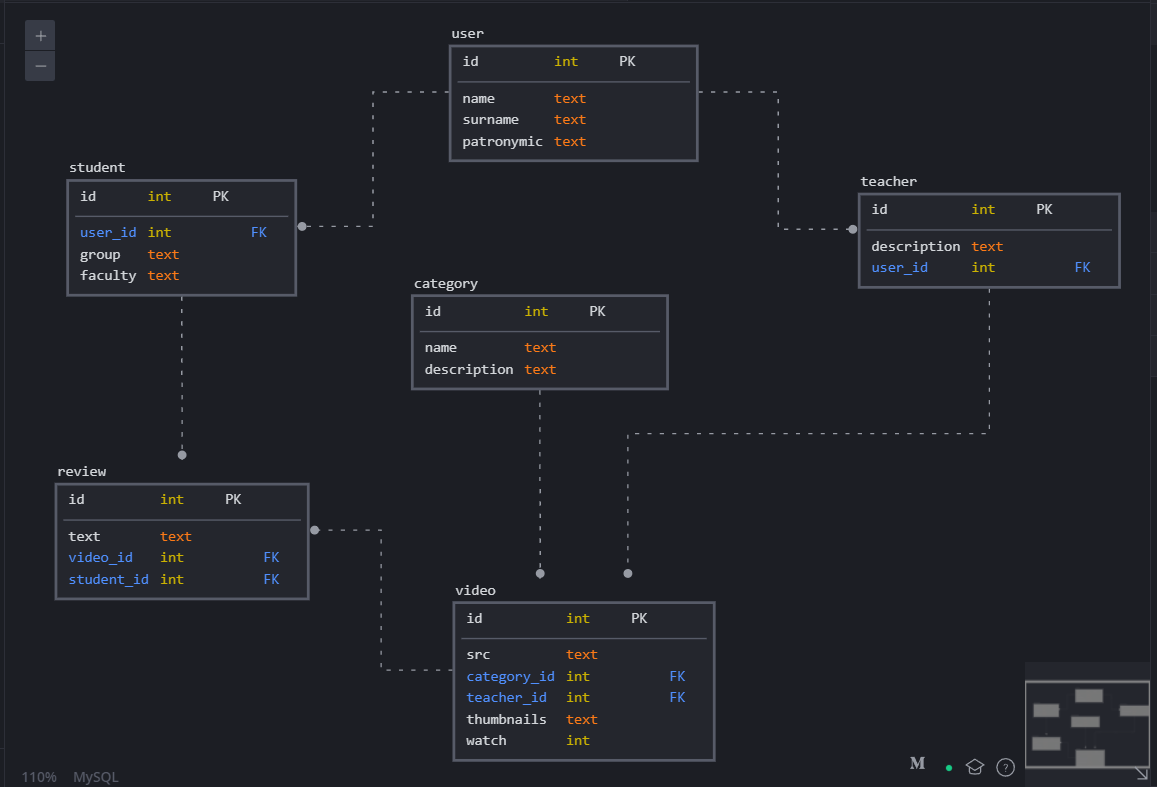
# Обосновать выбор средства разработки логической модели данных.

Для создания логической модели данных используется SQL Database Model так, как у SQL Database Model расширенный функционал для улучшения моделирования, удобный интерфейс и имеется возможность автоматически создать модель базы данных и внести изменения.

# Выполнить разработку логической модели данных.

Сущности

1. Пользователь (user)
2. Преподаватель (teacher)
3. Студент (student)
4. Курс (category)
5. Видеокурс (video)
6. Комментарии к видео (review)



Составим таблицу сущностей модели данных, в которой отражаются сущности, атрибуты и краткое описание сущностей.

|  |  |  |  |
| --- | --- | --- | --- |
| № | Сущность | Атрибуты | Описание |
| 1 | Пользователь | Id пользователя, имя, фамилия, отчество | Общие данные о пользователе |
| 2 | Студент | Id студента, Id пользователя, группа, факультет | Данные о студенте |
| 3 | Преподаватель | Id преподавателя, Id пользователя, описание | Данные о преподавателе |
| 4 | Категория | Id категории, название, описание | Данные о категории |
| 5 | Видео | Id видео, ссылка на видео, Id категории, Id преподавателя, превью, кол-во просмотров | Данные о видео |
| 6 | Отзыв | Id отзыва, отзыв, id видео, id студента | Данные о отзыве |

# Описать разработку физической модели данных.

Структура таблицы Пользователь

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Имя поля** | **Тип данных** | **Размер** | **Условие на значение** | **Значение по умолчанию** | **Примечание** |
| id | int | 255 | Not null | 0000 | ID пользователя |
| name | text | 255 | Not null |  | Имя |
| surname | text | 255 | Not null |  | Фамилия |
| patronymic | text | 255 | Not null |  | Отчество |

Структура таблицы Студент

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Имя поля** | **Тип данных** | **Размер** | **Условие на значение** | **Значение по умолчанию** | **Примечание** |
| id | int | 255 | Not null | 0000 | ID студента |
| User\_id | int | 255 | Not null | 0000 | ID пользователя |
| group | text | 255 | Not null |  | Группа |
| faculty | text | 255 | Not null |  | Факультет |

Структура таблицы Преподаватель

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Имя поля** | **Тип данных** | **Размер** | **Условие на значение** | **Значение по умолчанию** | **Примечание** |
| id | int | 255 | Not null | 0000 | ID преподавателя |
| User\_id | int | 255 | Not null | 0000 | ID пользователя |
| description | text | 255 | Not null |  | Описание |

Структура таблицы Категория

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Имя поля** | **Тип данных** | **Размер** | **Условие на значение** | **Значение по умолчанию** | **Примечание** |
| id | int | 255 | Not null | 0000 | ID категории |
| name | int | 255 | Not null | 0000 | Название |
| description | text | 255 | Not null |  | Описание |

Структура таблицы Видео

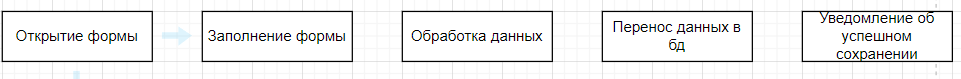
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Имя поля** | **Тип данных** | **Размер** | **Условие на значение** | **Значение по умолчанию** | **Примечание** |
| id | int | 255 | Not null | 0000 | ID видео |
| src | int | 255 | Not null | 0000 | Ссылка на видео |
| Category\_id | text | 255 | Not null |  | Id категории |
| Teacher\_id | int | 255 | Not null | 0000 | Id преподавателя |
| Thumbnails | text | 255 | Not null |  | Превью |
| Watch | int | 255 | Not null | 0000 | Кол-во просмотров |

Структура таблицы Отзывов

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Имя поля** | **Тип данных** | **Размер** | **Условие на значение** | **Значение по умолчанию** | **Примечание** |
| id | int | 255 | Not null | 0000 | ID отзыва |
| text | text | 255 | Not null |  | Текст отзыва |
| Video\_id | text | 255 | Not null |  | Id видео |
| Student\_id | text | 255 | Not null |  | Id студента |

# Описать процесс организации сбора и обработки информации.

В данной ИС используется информация о пользователе при регистрации и информация о курсах при создании курса.



Средства защиты БД от разрушения и несанкционированного доступа:

1. Защита с использованием пароля от несанкционированного доступа: будет добавлена регистрация пользователей.
2. Достойное качество программного обеспечения и разработки средств отражения угроз;
3. Назначение привилегии: будут добавлены роли (студент, методист и преподаватель).

# 6. Разработка математического обеспечения ИС

Алгоритм поиска

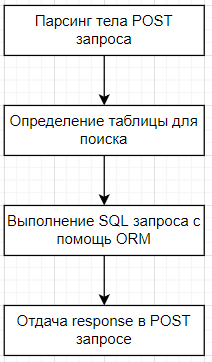
Назначения и характеристика: в нашей ИС используется алгоритм поиска, который выполняет условие нахождения нужной информации из БД и проверяя её наличие. Данный алгоритм будет обобщенным для пользователей, курсов и видео. Поиск будет происходить по полю name. Алгоритм поиска не сможет сработать если в поле формы для поиска пользователь будет вводить неправильную информацию.

Информация, которая используется: для того, чтобы алгоритм сработал используется информация о названии/имени, которую преподаватель или студент занесет в поле формы поиска на сайте, а также используется информация из таблицы БД «Пользователь», «Видео», «Категория».

Результаты решения: после того, как алгоритм сработал, пользователь сможет увидеть курсы, других пользователей или видео, которое он искал с помощью поиска на сайте.

Математическое описание: в данном алгоритме используется сравнение, где введенная информация заносится в массив и каждый элемент массива сравнивается с данными в БД. Если каждый элемент массива совпал с данными в БД, то нужная информация считается найденной.

Алгоритм решения: При поступлении POST запроса на сервер, обрабатывается тело запроса. В теле должно быть поле type, иначе берется по умолчанию type = video. В зависимости от поля type определяется таблица БД. «Пользователь» type = user, «Видео» type = video, «Категория» type = category. Далее делаем запрос в ORM (модуль для генерации sql запросов) для поиска данных в бд JSON форматом. Как пример: { type: user, name: “Алексан ”}, что значит сделать поиск в таблице «Пользователь» и выдать всех пользователей у которых поле name совпадает со строкой «Алексан»



# 7. Разработка программного обеспечения ИС

# Провести обоснованный выбор операционной системы для сервера и рабочих станций.

Для анализа операционных систем выберем две системы Windows и Linux.

Табл.1. Сравнение Windows и Linux

|  |  |  |
| --- | --- | --- |
| **Критерий** | **Windows** | **Linux** |
| **Стоимость** | От 199$ для бизнеса | Бесплатно |
| **Интерфейс и дизайн** | Красивый UI/UX | Устаревший дизайн, вне зависимости от дистрибутива |
| **Настройки** | Очень сложно настроить ПО под себя. | Гибко настраиваемая ПО |
| **Обновления** | Регулярно | Регулярно |
| **Установка программ** | Каталог приложений | Кателог приложений |
| **Безопасность** | Более уязвима к вирусам, т.к. является более популярной ПО | Менее уязвима к вирусам, т.к. является менее популярной ПО |
| **Производительность и стабильность** | Занимает достаточно много оперативной памяти | Стабильная быстрая скорость работы. |
| **Совместимость** | Обеспечивает совместимость с 97% всех выпускаемых игр. | Плохо совместима с приложениями. |
| **Кому из пользователей подходит** | Создана преимущественно для простых пользователей. | Простым пользователям и программистам. |

В нашей ИС будет использоваться операционная система Linux так, как он имеет больше возможностей, нежели Windows

# Провести обоснованный выбор средства разработки специального ПО.

Для средства разработки специального ПО сделаем выбор из двух систем Visual Studio Code и Atom.

В общем и целом VS Code и Atom имеют одинаковую функциональность, но Atom приобретается по подписке. Для создания нашей ИС функционала VS Code более чем достаточно, особенно с огромным выбором плагинов для редактора.

# Выбрать для обслуживания ИС сервисные программы и утилиты.

phpMyAdmin - Для отображения данных из БД.

Docker – для развертывания и управления приложением.

ISP manager – управление ПО хостинга и vps/vds сервера.

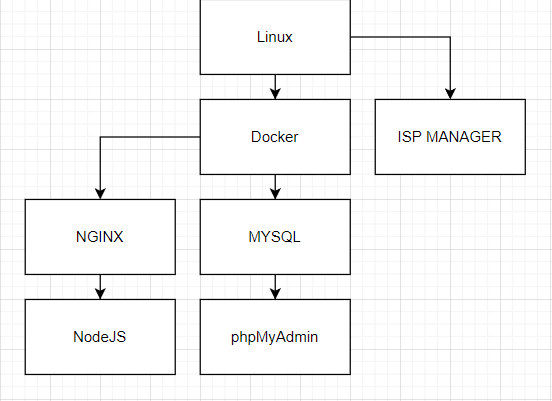
NodeJS – платформа для запуска серверной части

Nginx – веб сервер

# Разработать структуру специального программного обеспечения.

Описание основных функциональных модулей программы:

1. Модуль регистрации: пользователь заносит свои данные в поле для регистрации, затем информация о пользователе попадает в БД. Когда пользователь будет выполнять вход в систему, он будет заполнять поле входа в систему данными, которые уже есть в БД. Будет выполняться запрос к БД для проверки данных о пользователе. Если данные совпадают, то пользователь сможет войти в систему.
2. CRUD модуль видео: преподаватель загружает видео и вводит доп. информацию. CRUD подразумевает возможность: создания, чтения, редактирования и удаления
3. CRUD модуль отзыва: студент вводит отзыв к видео. CRUD подразумевает возможность: создания, чтения, редактирования и удаления
4. CRUD модуль Курса: преподаватель вводит доп. информацию к курсу. CRUD подразумевает возможность: создания, чтения, редактирования и удаления



# Составить и коротко описать схему взаимодействия БД и специального ПО.



# 8. Разработка технического обеспечения ИС

# Разработать решения по выбору конфигурации компьютеров для сервера и рабочих станций.

Первая сборка:

Процессор AMD FX-4300 OEM [AM3+, 4 x 3.8 ГГц, L2 - 4 МБ, L3 - 4 МБ, 2хDDR3-1866 МГц, TDP 95 Вт]

Материнская плата ASRock 760GM-HDV

Корпус GiNZZU B190 [17221] черный

Видеокарта MSI GeForce 210 [N210-1GD3/LP]

Кулер для процессора ID-COOLING DK-01T

Оперативная память AMD Radeon R5 Entertainment Series [R532G1601U1S-U] 2 ГБ

500 ГБ Жесткий диск WD Blue [WD5000AZRZ]

Блок питания AeroCool VX PLUS 350W [VX-350 PLUS]

Итоговая стоимость: 20 043 ₽

Вторая сборка:

Процессор Intel Core i5-11500 BOX [LGA 1200, 6 x 2.7 ГГц, L3 - 12 МБ, 2хDDR4-3200 МГц, Intel UHD Graphics 750, TDP 65 Вт, кулер]

Материнская плата ASRock Z590 Steel Legend

Корпус Sharkoon REV200 [REV200] черный

Видеокарта Powercolor AMD Radeon RX 6600 XT Hellhound OC [AXRX 6600XT 8GBD6-3DHL/OC]

Кулер для процессора Noctua NH-U12A [NH-U12A]

Оперативная память Kingston FURY Beast Black [KF436C18BBK4/128] 128 ГБ

2000 ГБ SSD M.2 накопитель Corsair Force Series MP600 [CSSD-F2000GBMP600]

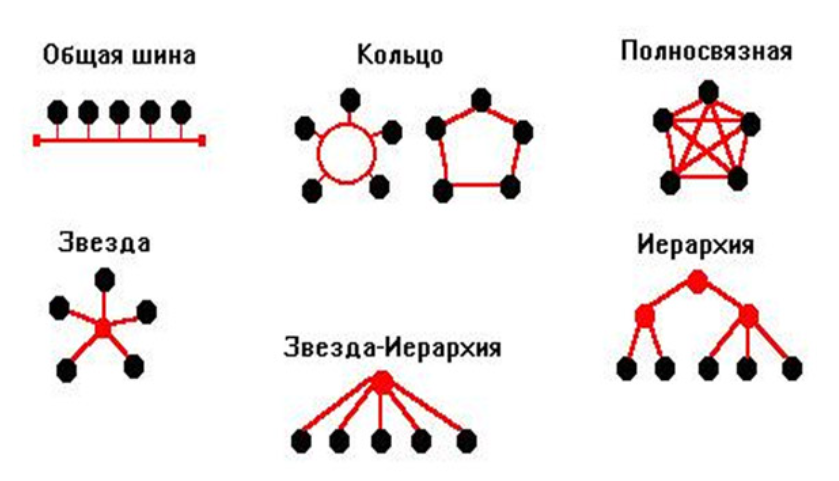
Блок питания Corsair RM1000x [CP-9020201-EU]

Итоговая стоимость: 231 292 ₽

Для наших задач достаточно первой сборки + она выйдет дешевле.

# Выбрать топологию компьютерной сети.

Термины топология сети, структура сети или конфигурация сети означают способ соединения компьютеров в сеть. Основные топологии: звезда, кольцо, шина.



**Топология “шина”**

Топология шина (или, как ее еще часто называют общая шина или магистраль) предполагает использование одного кабеля, к которому подсоединены все рабочие станции.

Общий кабель используется всеми станциями по очереди. Все сообщения, посылаемые отдельными рабочими станциями, принимаются и прослушиваются всеми остальными компьютерами, подключенными к сети. Из этого потока каждая рабочая станция отбирает адресованные только ей сообщения.

Достоинства топологии “шина”:

* простота настройки;
* относительная простота монтажа и дешевизна, если все рабочие станции расположены рядом;
* выход из строя одной или нескольких рабочих станций никак не отражается на работе всей сети.

Недостатки топологии “шина”:

* неполадки шины в любом месте (обрыв кабеля, выход из строя сетевого коннектора) приводят к неработоспособности сети;
* сложность поиска неисправностей;
* низкая производительность – в каждый момент времени только один компьютер может передавать данные в сеть, с увеличением числа рабочих станций производительность сети падает;
* плохая масштабируемость – для добавления новых рабочих станций необходимо заменять участки существующей шины.

**Топология “кольцо”**

Кольцо – это топология локальной сети, в которой рабочие станции подключены последовательно друг к другу, образуя замкнутое кольцо. Данные передаются от одной рабочей станции к другой в одном направлении (по кругу). Каждый ПК работает как повторитель, ретранслируя сообщения к следующему ПК, т.е. данные передаются от одного компьютера к другому как бы по эстафете.

Если компьютер получает данные, предназначенные для другого компьютера – он передает их дальше по кольцу, в ином случае они дальше не передаются.

* Достоинства кольцевой топологии:
* простота установки;
* практически полное отсутствие дополнительного оборудования;
* возможность устойчивой работы без существенного падения скорости передачи данных при интенсивной загрузке сети.

Однако “кольцо” имеет и существенные недостатки:

* каждая рабочая станция должна активно участвовать в пересылке информации; в случае выхода из строя хотя бы одной из них или обрыва кабеля – работа всей сети останавливается;
* подключение новой рабочей станции требует краткосрочного выключения сети, поскольку во время установки нового ПК кольцо должно быть разомкнуто;
* сложность конфигурирования и настройки;
* сложность поиска неисправностей.

Кольцевая топология сети используется довольно редко. Основное применение она нашла в оптоволоконных сетях стандарта Token Ring.

**Топология “звезда”**

Звезда – это топология локальной сети, где каждая рабочая станция присоединена к центральному устройству (коммутатору или маршрутизатору). Центральное устройство управляет движением пакетов в сети. Каждый компьютер через сетевую карту подключается к коммутатору отдельным кабелем.

Топология “звезда” на сегодняшний день стала основной при построении локальных сетей. Это произошло благодаря ее многочисленным достоинствам:

* выход из строя одной рабочей станции или повреждение ее кабеля не отражается на работе всей сети в целом;
* отличная масштабируемость: для подключения новой рабочей станции достаточно проложить от коммутатора отдельный кабель;
* легкий поиск и устранение неисправностей и обрывов в сети;
* высокая производительность;
* простота настройки и администрирования;
* в сеть легко встраивается дополнительное оборудование.

Однако, как и любая топология, “звезда” не лишена недостатков:

* выход из строя центрального коммутатора обернется неработоспособностью всей сети;
* дополнительные затраты на сетевое оборудование – устройство, к которому будут подключены все компьютеры сети (коммутатор);
* число рабочих станций ограничено количеством портов в центральном коммутаторе.

Вывод: Для нашей ИС мы будем использовать топологию «Звезда», т.к. она является самой распространенной топологией для проводных и беспроводных сетей, а значит поиск гайдов и различных статей по настройке будет упрощен.

# Выполнить выбор прочих технических устройств, задействованных в ИС.

Будет использован кабель Gigabit Ethernet для высокоскоростного интернет подключения, источник бесперебойного питания CyberPower BS450E.

# Разработать структурную схему комплекса технических средств (КТС).

К серверу, то есть главному компьютеру подключен источник бесперебойного питания. Сам сервер подключен к сети интернет, а подключенные пользователи к сети интернет подключаются к нашему серверу, чтобы воспользоваться сайтом.



# 9. Анализ рисков внедрения и эксплуатация ИС

# Выявить потенциальные риски для проектируемой информационной системы (не менее 9).

|  |  |  |  |
| --- | --- | --- | --- |
| № п.п. | Риск | Категория | Последствия риска |
| 1 | Нехватка финансирования для MVP | В | ИС не начнет работать даже на стадии разработки |
| 2 | Нехватка тех. мощностей | В | Производительность ИС будет на низком уровне |
| 3 | Отказ развертывания ИС | В | При отказе развертывания ИС, в продакшене будет старая версия ИС |
| 4 | Отключение интернета провайдером | В | Из-за высокой нагруженности на сеть, провайдер может отключить нам доступ к интернету |
| 5 | Масштабируемость | Э | Производительность ИС будет на низком уровне |
| 6 | Проникновение в ИС сторонних лиц | Э | Может привести к изменению контента и выводу непредназначенного для данного сайта материала |
| 7 | Скорость работы ИС | Э | Приведет к тому, что пользователи перестанут пользоваться ИС |
| 8 | Финансы | Э | Для функционирования ИС, нужен постоянный денежный приток. (К примеру аренда серверов) |
| 9 | Непредвиденные риски | Э | Риски о которых мы узнаем только на практике при запуске ИС |

# Провести оценивание рисков, используя шкалы оценивания тяжести и вероятности реализации.

Классификация уровней профессионального риска



Шкала оценивания вероятности реализации риска

|  |  |  |
| --- | --- | --- |
| № п.п. | Вероятность реализации | Диапазон вероятности реализации |
| 1 | 1 | 0-20 |
| 2 | 2 | 0-20 |
| 3 | 3 | 0-20 |
| 4 | 4 | 0-20 |
| 5 | 2 | 21-70 |
| 6 | 3 | 71-200 |
| 7 | 1 | 21-70 |
| 8 | 3 | 21-70 |
| 9 | 1 | 71-200 |

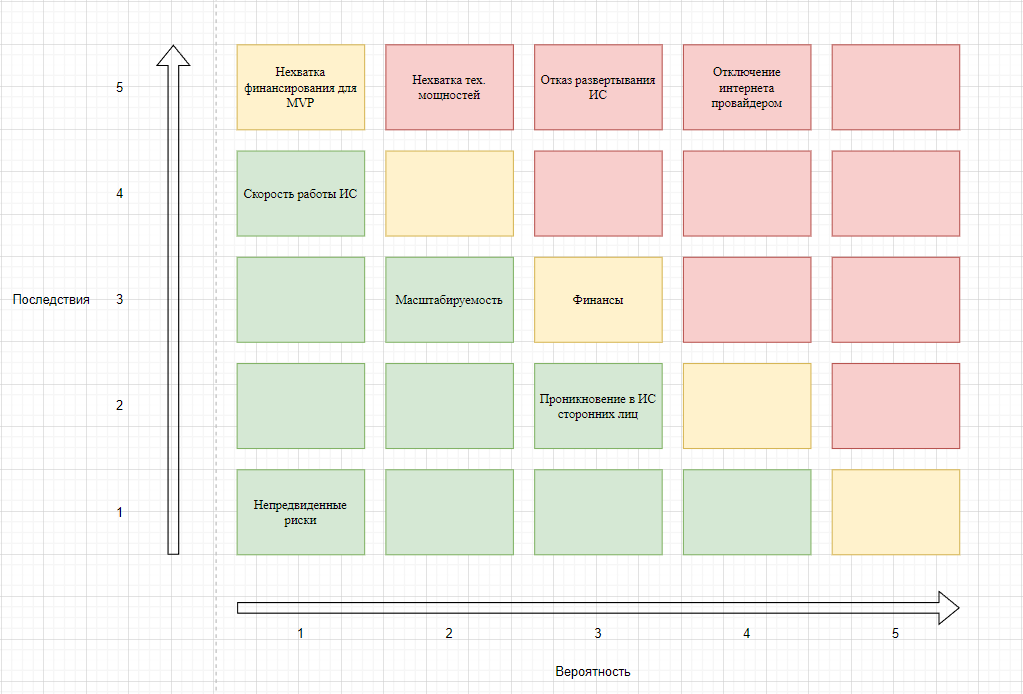
Шкала оценивания существенности последствий риска

|  |  |  |
| --- | --- | --- |
| № п.п. | Оценка существенности | Существенность последствий |
| 1 | 5 | Очень высокая |
| 2 | 5 | Очень высокая |
| 3 | 5 | Очень высокая |
| 4 | 5 | Очень высокая |
| 5 | 3 | Средняя |
| 6 | 2 | Низкая |
| 7 | 5 | Очень высокая |
| 8 | 3 | Средняя |
| 9 | 1 | Очень низкая |

Соотнесение риска с вероятностью реализации и существенностью последствий:

|  |  |  |  |
| --- | --- | --- | --- |
| № п.п. | Риск | Вероятность реализации риска | Существенность последствий риска |
| 1 | Нехватка финансирования для MVP | 1 | Очень высокая |
| 2 | Нехватка тех. мощностей | 2 | Очень высокая |
| 3 | Отказ развертывания ИС | 3 | Очень высокая |
| 4 | Отключение интернета провайдером | 4 | Очень высокая |
| 5 | Масштабируемость | 2 | Средняя |
| 6 | Проникновение в ИС сторонних лиц | 3 | Низкая |
| 7 | Скорость работы ИС | 1 | высокая |
| 8 | Финансы | 3 | Средняя |
| 9 | Непредвиденные риски | 1 | Очень низкая |

# Заполнить карту рисков на основе выявленных ранее рисков и шкал оценивания.



# Провести ранжирование рисков.

На первом месте для решения будут располагаться риски из области критических рисков, к таким рискам относятся:

1. Нехватка тех. мощностей;
2. Отказ развертывания ИС;
3. Отключение интернета провайдером;

Второе место занимают риски, лежащие в области умеренных потерь, к таким рискам относятся:

1. Нехватка финансирования для MVP;
2. Финансы;

На третьем месте будут располагаться риски, которые являются управляемыми в рабочем режиме, к таким рискам относятся:

1. Скорость работы ИС;
2. Масштабируемость;
3. Проникновение в ИС сторонних лиц;
4. Непредвиденные риски

# 9.5 На основании полученных данных сделать заключение о целесообразности внедрения информационной системы на объекте.

Исходя из всего вышеперечисленного, можно сделать вывод, что риски которые попадают в зону высокого риска, находятся под категорией «В», а значит мы столкнемся с ними один раз, исправив проблему, она больше не возникнет. Риски, которые попадают в зону умеренного риска, относятся к финансовой деятельности, а значит для старта/поддержания проекта нужны минимальные вложения. Риски, которые находятся в зоне низкого риска, являются штатными и их можно решить после релиза продукта.