**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ**

**КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Поволжский государственный университет телекоммуникаций и информатики»

Кафедра ИСТ

Факультет ИСТ

**ДОПУСТИТЬ К ЗАЩИТЕ ЗАЩИЩЕНА С ОЦЕНКОЙ** \_\_\_\_

Руководитель \_\_\_\_\_\_\_\_\_\_ Бедняк С.Г.                         Руководитель \_\_\_\_\_\_\_\_\_ Бедняк С.Г.   
                        (подпись)      (ФИО)                                                           (подпись)    (ФИО)

       \_\_\_\_\_\_\_\_\_\_\_  202 г.                                                             \_\_\_\_\_\_\_\_\_\_\_ 202 г.   
  (дата)                                                                                      (дата) 

**КУРСОВАЯ РАБОТА**

по дисциплине «Платформы и программные средства разработки информационных систем

Проектирование и разработка ИС для учебной части колледжа

Вариант 5

Выполнил

Студент гр. ИСТ-25у

Часовских Григорий Сергеевич

№ зачетной книжки 220241

Самара 2023

**Рецензия**

Курсовая работа студента Часовских Г.С. на тему «Проектирование и разработка ИС для учебной части колледжа» выполнена в соответствии с заданием, соблюдены все требования, предъявляемые к курсовой работе. По заданной предметной области создана база данных и приложение для работы с ней.

Теоретическая часть представлена в достаточном объёме. Однако, можно было лучше раскрыть теоретическую часть и реализовать некоторые программные решения, используя другие методы.

Считаю, что курсовая работа выполнена на высоком уровне. Тема курсовой работы раскрыта полностью, задание выполнено в полном объёме, выводы сформулированы студентом самостоятельно.

Часовских Г.С. допущен к защите. Рекомендуемая оценка –

**Руководитель Бедняк С.Г.**

**Содержание**

[Введение 4](#_Toc153978983)

[1 Теоретические основы разработки 5](#_Toc153978984)

[1.1 Анализ предметной области 7](#_Toc153978985)

[2 Выбор языка, среды программирования и программ 9](#_Toc153978986)

[3 Жизненный цикл информационной системы 10](#_Toc153978987)

[4 Создание базы данных 12](#_Toc153978988)

[5 Создание приложения 16](#_Toc153978989)

[6 Обзор готового приложения 19](#_Toc153978990)

[7 Запуск одноузлового Kubernetes-кластера с помощью Minikube 21](#_Toc153978991)

[8 Запуск кластера с помощью Kubeadm 25](#_Toc153978992)

[Заключение 28](#_Toc153978993)

[Список использованных источников 29](#_Toc153978994)

[Приложение А - Техническое задание 30](#_Toc153978995)

[Приложение B - Эскизный проект 34](#_Toc153978996)

[Приложение C - Технический проект 40](#_Toc153978997)

# Введение

В настоящее время все больше предприятий сталкиваются с вызовом повышения эффективности управления: усиление контроля и ускорение хода рабочих процессов, улучшение возможности их мониторинга, оптимизация использования рабочего времени, сокращение трудозатрат, увеличение производительности труда и так далее. Единственным способом достижения таких целей является внедрение информационной системы.

В данной работе будет создана ИС для учебной части колледжа. Данная ИС позволит работникам колледжа выполнять свои функции гораздо проще и быстрее, тем самым повысив их продуктивность труда.

Актуальность темы определяется тем, что данный колледж не имеет ИС для эффективного повышения производительности, поэтому рабочие процессы протекают не так эффективно, как хотелось бы.

Целью данной курсовой работы является разработка информационной системы для учебной части колледжа “СамПГУ”, изучение и применение на практике знаний в области разработки информационных систем, полученных в процессе изучения курса по дисциплине “Платформы и программные среды разработки информационных систем”.

# 1 Теоретические основы разработки

В современном мире практически любая автоматизация какого-либо процесса или предприятия связана с разработкой информационной системы, которая в свою очередь использует цифровые технологии. Наступившая цифровая эпоха связана с большими объемами данных, которые необходимо хранить в компактном виде и эффективно обрабатывать. Для создания автоматизированных систем используют проектирование.

Проектирование — процесс разработки проекта, то есть комплекта документации, предназначенной для создания определённого объекта, его эксплуатации, ремонта и ликвидации, а также для проверки или воспроизведения промежуточных и конечных решений, на основе которых был разработан данный объект. Проектирование – длительный процесс и включает этапы от подготовки технического задания до испытания опытных образцов. Проектирование, независимо от его содержания, это составная часть планирования Объектом проектирования может быть материальный предмет, выполнение работы, оказание услуги.

Информационная система – это совокупность технических, организационных средств, программного обеспечения, персонала и систем баз данных предназначенные для своевременного обеспечения информационных потребностей людей.

Жизненный цикл информационной системы — это процесс разработки, развития и эксплуатации информационной системы начинающийся с момента принятия решения о создании в плоть до момента вывода её из эксплуатации.

«Python»[5] — высокоуровневый язык программирования общего назначения с динамической строгой типизацией и автоматическим управлением памятью, ориентированный на повышение производительности разработчика, читаемости кода и его качества, а также на обеспечение переносимости написанных на нём программ. Язык является полностью объектно-ориентированным в том плане, что всё является объектами. Необычной особенностью языка является выделение блоков кода отступами. Синтаксис ядра языка минималистичен, за счёт чего на практике редко возникает необходимость обращаться к документации. Сам же язык известен как интерпретируемый и используется в том числе для написания скриптов.

«Pymysql»[2] — это библиотека для работы с базой данных MySQL на языке программирования Python. Она предоставляет удобные средства для взаимодействия с MySQL, позволяя выполнять запросы, получать результаты и управлять базой данных прямо из Python-скриптов.

База данных[4] — упорядоченный набор структурированной информации или данных, которые обычно хранятся в электронном виде в компьютерной системе.

Проектирование базы данных — это процесс создания проекта базы данных, предназначенной для поддержки функционирования экономического объекта и способствующей достижению его целей.

Методология проектирования может рассматриваться как совокупность методов и

средств, последовательное применение которых обеспечивает разработку проекта базы

данных, удовлетворяющего целям проектирования. Задачей процесса проектирования

базы данных, адекватно отражающей описываемую проблемную среду, реализуемой с помощью технических и программных средств. При создании эффективного приложения, работающего с информацией, хранящейся в БД, основное внимание должно быть уделено проектированию такой структуры базы данных, которая должна обеспечивать выполнение следующих требований:

1. Обеспечение целостности БД;
2. Мобильность прикладного программного обеспечения;
3. Простой и понятный пользователю ввод информации в базу данных и её

обновление;

1. Быстрое нахождение и выборка требуемой информации:
2. Защищенность от случайных сбоев и несанкционированного доступа.

Процесс проектирования состоит в переходе от одного уровня абстракции в представлении данных к другому. Этот процесс представляют последовательностью более простых, обычно интерактивных процессов проектирования менее сложных отображений между промежуточными моделями данных.

**1.1 Анализ предметной области**

Колледж – это учреждение, которое занимается обучением студентов различным специальностям. Специальности колледжа могут быть различными, в зависимости от его направленности. Однако, основными задачами колледжа являются:

1. Образование и обучение.
2. Подготовка к профессиональной деятельности.
3. Научно-исследовательская работа.
4. Развитие студенческой личности.
5. Содействие социальной мобильности.
6. Партнерство с бизнесом и обществом.
7. Контроль качества образования.

При проектировании информационной системы для колледжа необходимо учитывать все эти задачи и потребности заказчика[A,B,C]. Кроме того, необходимо учитывать требования законодательства, например, по защите персональных данных.

Давайте представим вымышленное предприятие под названием «СамПГУ». Это колледж, который предлагает широкий спектр обучения различным специальностям для жителей города Самары.

Целью разработки информационной системы для колледжа «СамПГУ» является повышение количества людей, имеющих высшее образование и улучшение качества образовательных программ.

В колледжа «СамПГУ» будут решаться следующие задачи:

1. Обеспечение образовательных программ, соответствующих актуальным требованиям рынка труда.
2. Поддержка студентов в процессе обучения и подготовке к трудоустройству.
3. Предоставление возможностей для профессионального развития и обучения студентов.
4. Обеспечение доступности образования для различных социальных и экономических групп населения.
5. Привлечение и подготовка квалифицированных преподавателей.

От нас, как разработчиков информационной системы, заказчик будет ожидать следующее:

1. **Эффективность**: Система должна быть быстрой и эффективной, чтобы обеспечить гладкую работу колледжа.
2. **Безопасность данных:** Система должна обеспечивать безопасное хранение и передачу данных о сотрудниках и студентах.
3. **Удобство использования**: Система должна быть простой в использовании для сотрудников колледжа и других пользователей.
4. **Интеграция с другими системами**: Система должна легко интегрироваться с другими системами.
5. **Поддержка и обслуживание**: Мы должны предоставлять надежную техническую поддержку и обслуживание системы после ее внедрения.

Разработка информационной системы для колледжа позволит улучшить организацию работы учреждения.

Эффект от внедрения информационной системы будет заключаться в следующем:

1. Ускорение процесса добавления студентов и преподавателей в систему.
2. Улучшение контроля за обучаемостью студентов.
3. Увеличение точности и актуальности информации о студентах и преподавателях.
4. Упрощение работы сотрудникам колледжа.

Таким образом, разработка информационной системы для колледжа будет полезна как для сотрудников учреждения, так и для его студентов, она значительно улучшит эффективность работы колледжа и качество обучения.

# 2 Выбор языка, среды программирования и программ

В данном проекте предстоит проектировать базу данных и создавать приложение для работы с ней, поэтому необходимо определиться со средствами проектирования и отладки. Отличным решением является пакет программ «Open Server». Он имеет в своём наборе «phpMyAdmin», который будет использоваться для проектирования базы данных.

«Open Server»[3] — это портативный локальный WAMP/WNMP сервер, имеющий многофункциональную управляющую программу и большой выбор подключаемых компонентов. Представленный пакет программ не является очередной любительской сборкой, собранной «на коленке», это первый полноценный профессиональный инструмент, созданный специально для веб-разработчиков с учётом их рекомендаций и пожеланий.

«phpMyAdmin»[8] — веб-приложение с открытым кодом, написанное на языке PHP и представляющее собой веб-интерфейс для администрирования СУБД MySQL. PhpMyAdmin позволяет через браузер и не только осуществлять администрирование сервера MySQL, запускать команды SQL и просматривать содержимое таблиц и баз данных. Приложение пользуется большой популярностью у веб-разработчиков, так как позволяет управлять СУБД MySQL без непосредственного ввода SQL команд.

SQL[6] - язык запросов, который применяют, чтобы работать с базами данных, структурированных особым образом.

Для написания кода приложения на языке Python использовалась среда разработки PyCharm — кроссплатформенная интегрированная среда разработки для языка программирования Python, разработанная компанией JetBrains на основе IntelliJ IDEA. Предоставляет пользователю комплекс средств для написания кода и визуальный отладчик. PyCharm поддерживает использование плагинов для облегчения написания кода. Пользователи могут сами писать свои плагины, тем самым расширять возможности PyCharm. Некоторые плагины из других JetBrains IDE могут работать с PyCharm.

# 3 Жизненный цикл информационной системы

Жизненный цикл информационной системы – период времени, который начинается с момента принятия решения о необходимости создания информационной системы заканчивается в момент ее полного изъятия из эксплуатации. Понятие жизненного цикла (ЖЦ) является одним из базовых понятий методологии проектирования информационных систем.

Модель жизненного цикла ИС — структура, определяющая последовательность выполнения и взаимосвязи процессов, действий и задач на протяжении жизненного цикла. Модель жизненного цикла зависит от специфики, масштаба и сложности проекта и специфики условий, в которых система создается и функционирует.

Модель ЖЦ ИС включает в себя:

* стадии;
* результаты выполнения работ на каждой стадии;
* ключевые события — точки завершения работ и принятия решений.

В качестве модели жизненного цикла ИС была выбрана спиральная модель.

Спиральная модель[1] (рисунок 3.1) предполагает постепенное развитие прототипов информационной системы. Каждый “виток” спирали соответствует созданию работоспособного фрагмента или версии системы.

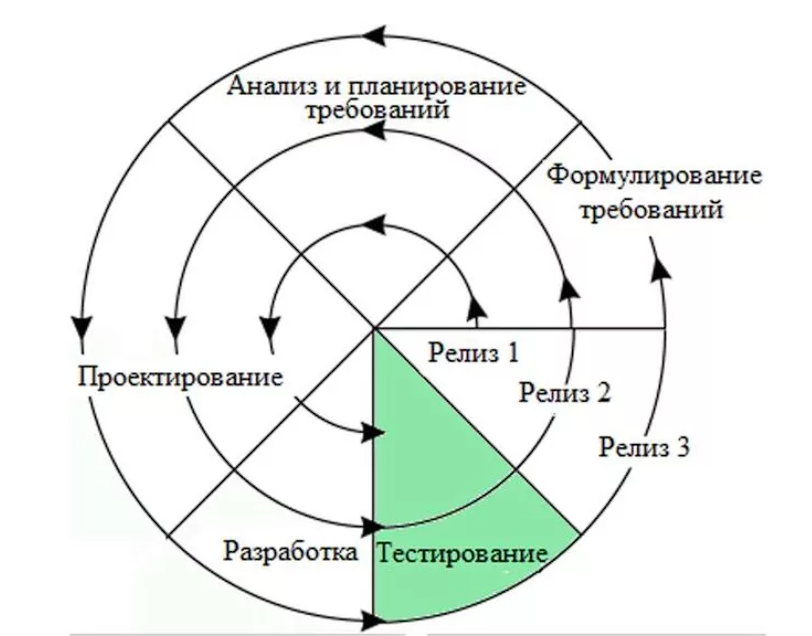


Рисунок 3.1 - Спиральная модель

Достоинства спиральной модели:

* Применяется при комплексной автоматизации, обеспечивая постепенную разработку и наращивания прототипа, уточнение требований по ходу разработки;
* Ускорение процесса разработки, проверка реализации требования по ходу разработки, отработка интерфейсов между компонентами системы.
* Недостатки спиральной модели:
* Увеличение объема работ по программированию требует применения специальных средств автоматизированного проектирования.

Данная модель подойдёт для разработки информационной системы для учебной части колледжа, так как позволит в дальнейшем реализовывать новый функционал и дорабатывать имеющийся. Данная курсовая работа опишет первый виток спиральной модели, одновременно с этим заложив фундамент дальнейшей разработки программного обеспечения и информационной системы предприятия колледж «СамПГУ».

# 4 Создание базы данных

На данном этапе разработки информационной системы происходит создание базы данных. Создаётся на основе эскизного проекта СУБД «СамПГУ». В качестве инструмента разработки базы данных будет использован интерфейс системного администратора, встроенного в open server – phpMyAdmin.

Запрос создания таблицы «Дисциплины»:

CREATE TABLE `disciplines` (

`id\_dis` int NOT NULL,

`id\_gryp` int NOT NULL,

`id\_teacher` int NOT NULL,

`name` varchar(30) NOT NULL,

`prikas\_form` varchar(30) DEFAULT NULL,

`year\_form` year DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

Запрос создания таблицы «Группы»:

CREATE TABLE `gryppa` (

`id\_gryp` int NOT NULL,

`id\_spec` int NOT NULL,

`name` varchar(10) DEFAULT NULL,

`type` varchar(20) DEFAULT NULL,

`kurs` tinyint UNSIGNED DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

Запрос создания таблицы «Кафедры»:

CREATE TABLE `kafedra` (

`id\_kaf` int NOT NULL,

`name` varchar(30) NOT NULL,

`head` varchar(30) DEFAULT NULL,

`priras\_form` varchar(30) DEFAULT NULL,

`year\_form` year DEFAULT NULL,

`adres` varchar(30) DEFAULT NULL,

`telephone` varchar(14) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

Запрос создания таблицы «Специальности»:

CREATE TABLE `specialnost` (

`id\_spec` int NOT NULL,

`name` varchar(50) CHARACTER SET utf8mb4 COLLATE utf8mb4\_0900\_ai\_ci NOT NULL,

`year\_form` year DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

Запрос создания таблицы «Студенты»:

CREATE TABLE `student` (

`id\_stud` int NOT NULL,

`id\_gryp` int NOT NULL,

`surname` varchar(30) NOT NULL,

`name` varchar(30) NOT NULL,

`seсname` varchar(30) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

Запрос создания таблицы «Личная карта студента»:

CREATE TABLE `studentcard` (

`id\_stud` int NOT NULL,

`year\_postup` year DEFAULT NULL,

`prikas\_postup` text,

`stependia` int DEFAULT NULL,

`birthday` date DEFAULT NULL,

`adres` text,

`email` varchar(30) DEFAULT NULL,

`telephone` varchar(14) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

Запрос создания таблицы «Преподаватели»:

CREATE TABLE `teacher` (

`id\_teach` int NOT NULL,

`id\_kaf` int NOT NULL,

`surname` varchar(30) NOT NULL,

`name` varchar(30) NOT NULL,

`secname` varchar(30) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

Запрос создания таблицы «Личная карта преподавателя»:

CREATE TABLE `teachercard` (

`id\_teach` int NOT NULL,

`year\_hired` year DEFAULT NULL,

`adress` text CHARACTER SET utf8mb4 COLLATE utf8mb4\_0900\_ai\_ci,

`email` varchar(30) DEFAULT NULL,

`telephone` varchar(14) DEFAULT NULL,

`wages` int UNSIGNED DEFAULT NULL,

`study\_type` varchar(30) DEFAULT NULL,

`study\_year` year DEFAULT NULL,

`birthday` date DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

После создания таблиц, их необходимо связать. Для этого используется инструмент phpMyAdmin под названием дизайнер. С его помощью в графическом режиме возможно создавать связи (рисунок 4.1)

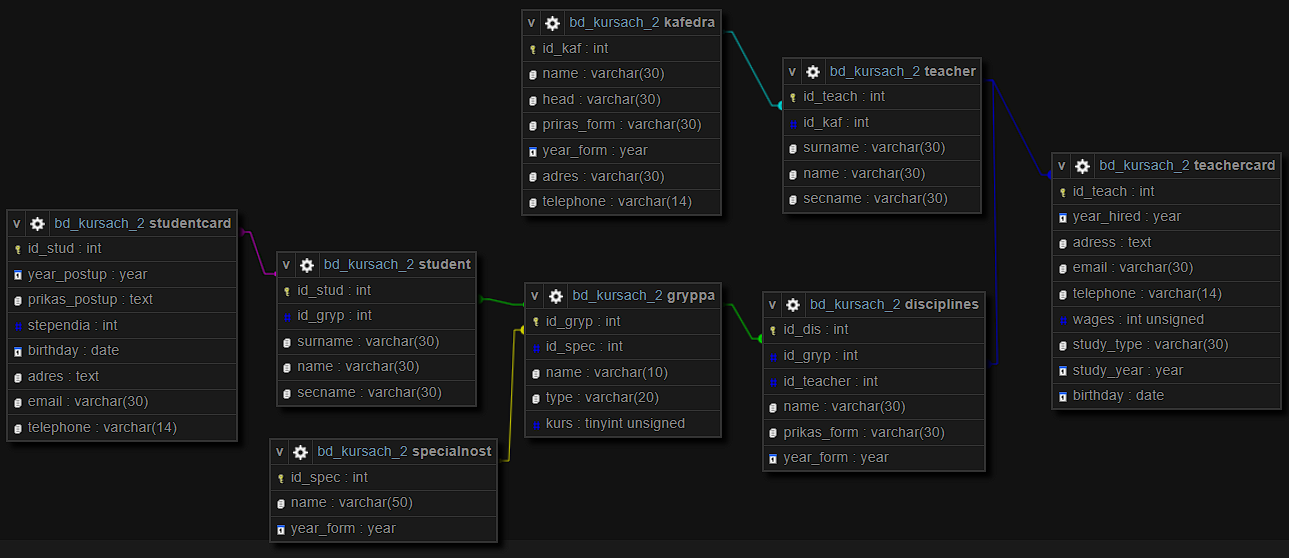


Рисунок 4.1 – ER – диаграмма базы данных

После создания связей необходимо внести данные в каждую из таблиц (рисунки 4.2-4.9) для обеспечения работы базы данных и функционала программного обеспечения.



Рисунок 4.2 – данные таблицы «Дисциплины»

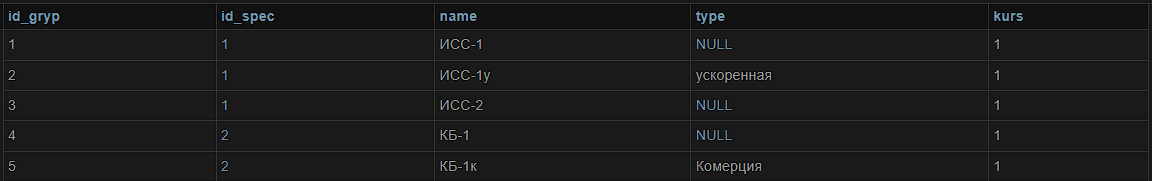


Рисунок 4.3 – данные таблицы «Группы»



Рисунок 4.4 – данные таблицы «Кафедры»

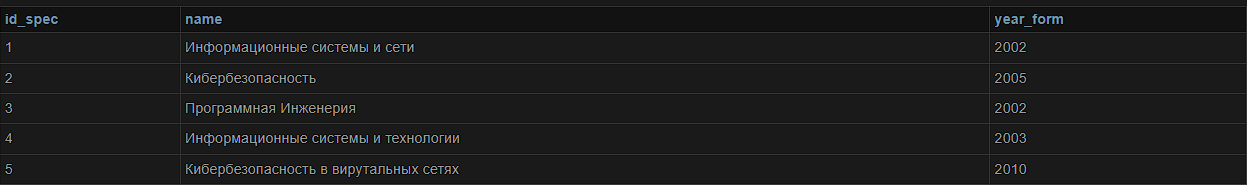


Рисунок 4.5 – данные таблицы «Специальности»



Рисунок 4.6 – данные таблицы «Студенты»



Рисунок 4.7 – данные таблицы «Личные карты студентов»

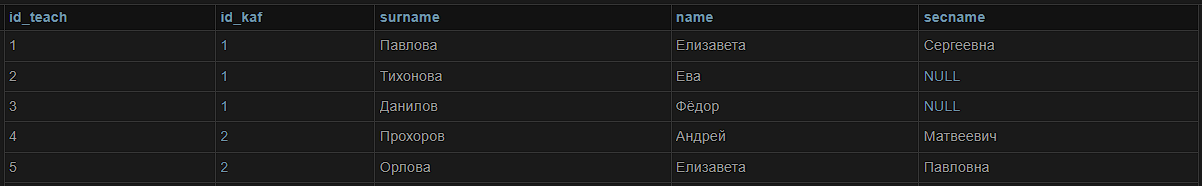


Рисунок 4.8 – данные таблицы «Преподаватели»

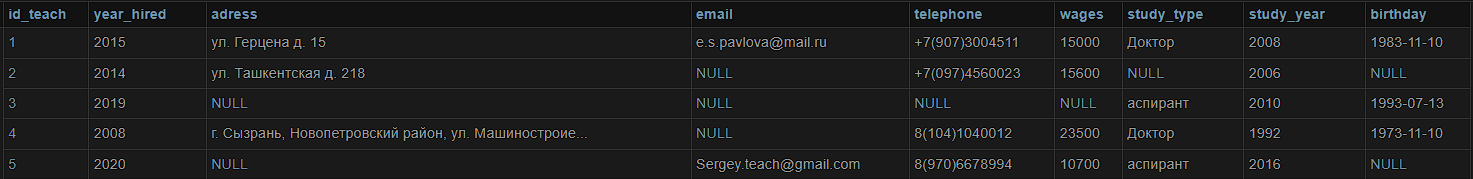


Рисунок 4.9 – данные таблицы «Личные карты преподавателей»

# 5 Создание приложения

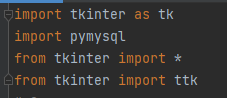


Рисунок 5.1 - импорт библиотек

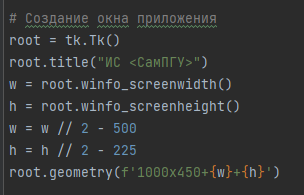


Рисунок 5.2 - Задание наименования и размера окна приложения

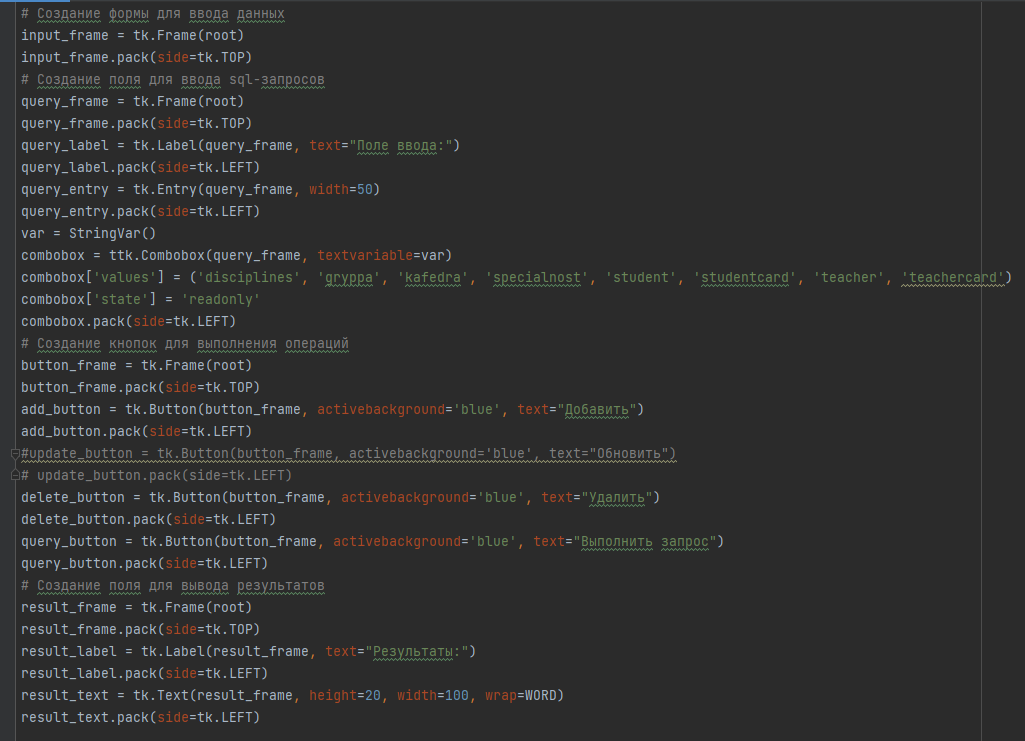


Рисунок 5.3 - инициализация элементов интерфейса (кнопки, строка для ввода запроса, таблицы с результатом запроса)

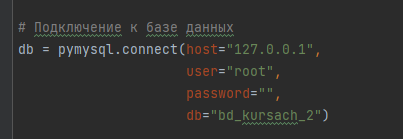


Рисунок 5.4 – установка связи с базой данных

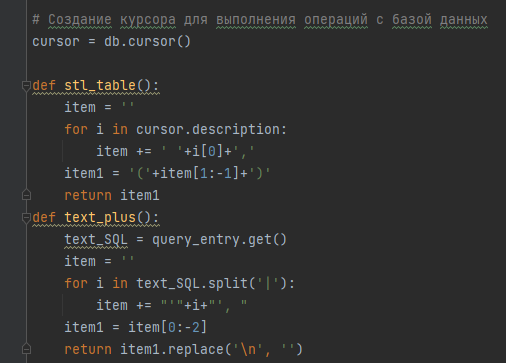


Рисунок 5.5 - Создание курсора для выполнения операций с базой данных

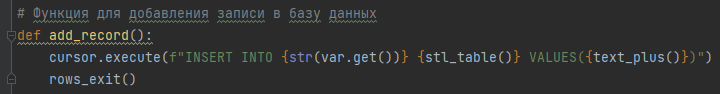


Рисунок 5.6 - Функция для добавления записи в базу данных

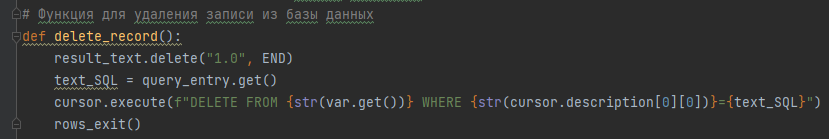


Рисунок 5.7 - Функция для удаления записи из базы данных

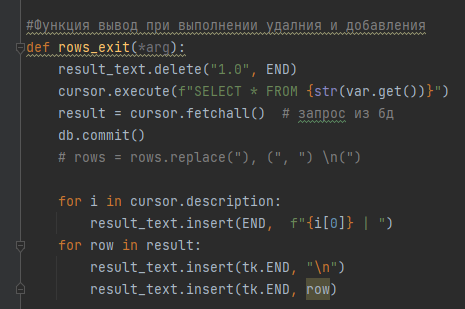


Рисунок 5.8 - Функция вывод при выполнении удаления и добавления

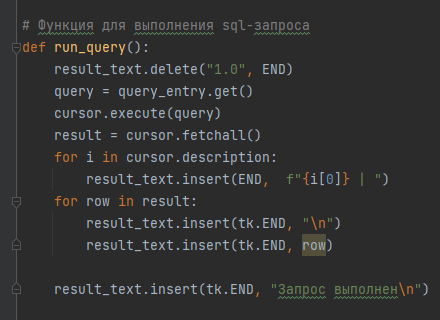


Рисунок 5.9 - Функция для выполнения sql-запроса

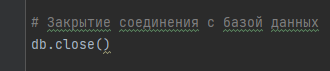


Рисунок 5.9 - Закрытие соединения с базой данных

# 6 Обзор готового приложения

Внешний вид оконного приложения:

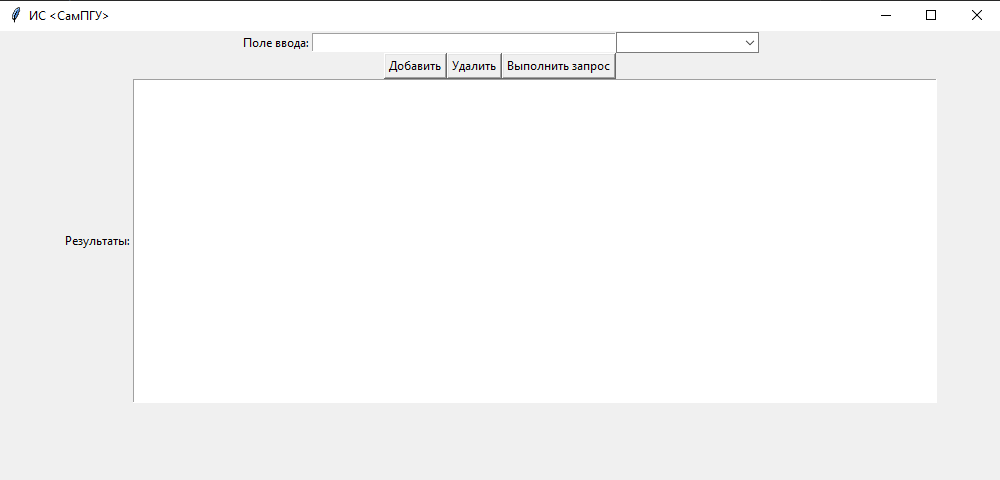
****

Рисунок 6.1 - вид программы при запуске

Внешний вид оконного приложения таблицы «Студенты»:

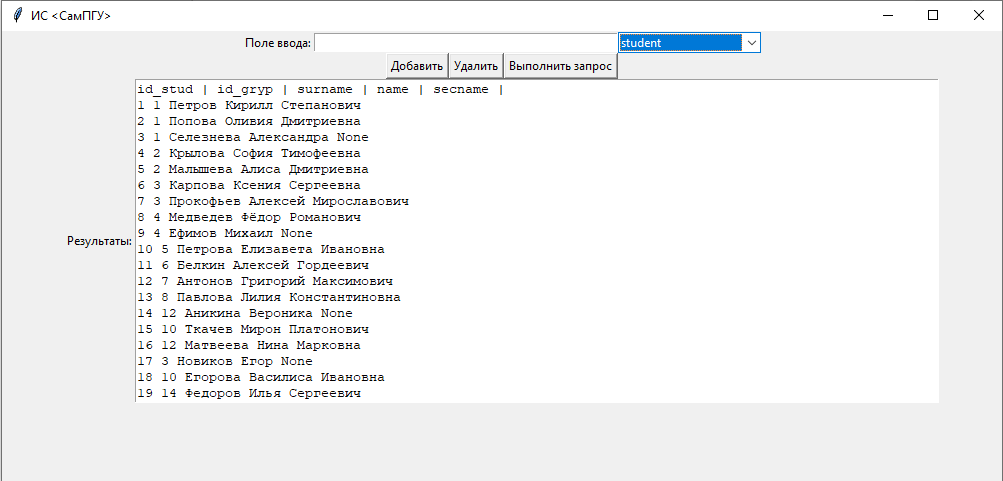


Рисунок 6.2 - вывод данных из таблицы "Студенты" при помощи программы

Добавление данных в БД.

1. Выбираем таблицу куда нужно добавить информацию.
2. Вводим в поле нужную информацию в формате 33|8|Грачев|Мирон|Романович

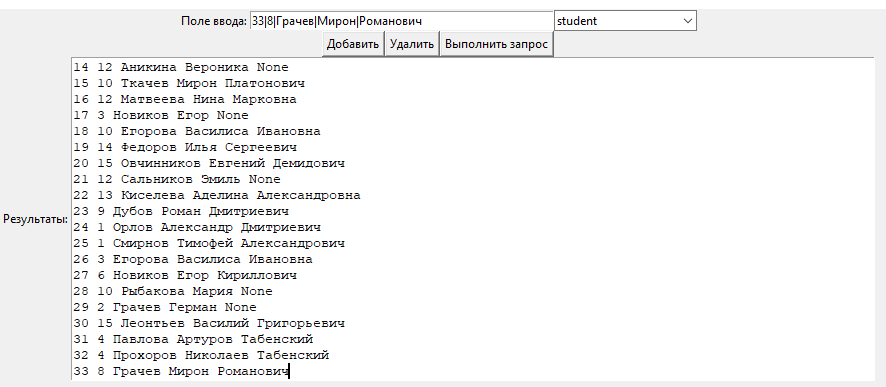


Рисунок 6.3 – функция ввода новых данных в таблицу

Удаление выполняется путем вписывания id строки который мы хотим удалить.

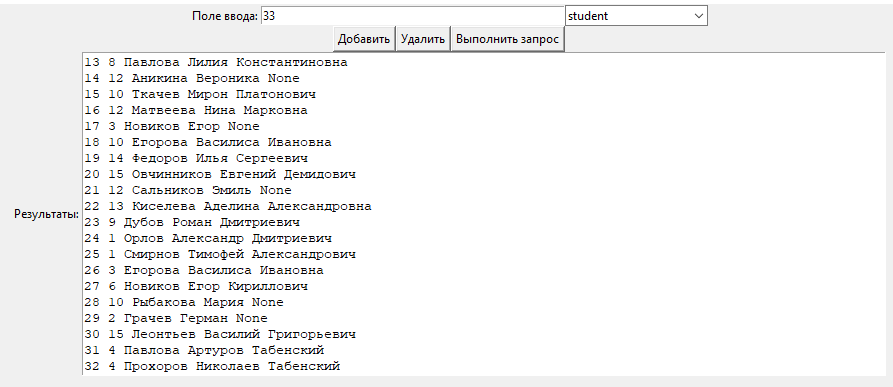


Рисунок 6.5 - функция удаление данных из таблицы

Выполнения SQL-запросов происходит при введение его в поле вывода и нажатия на кнопку выполнить запрос

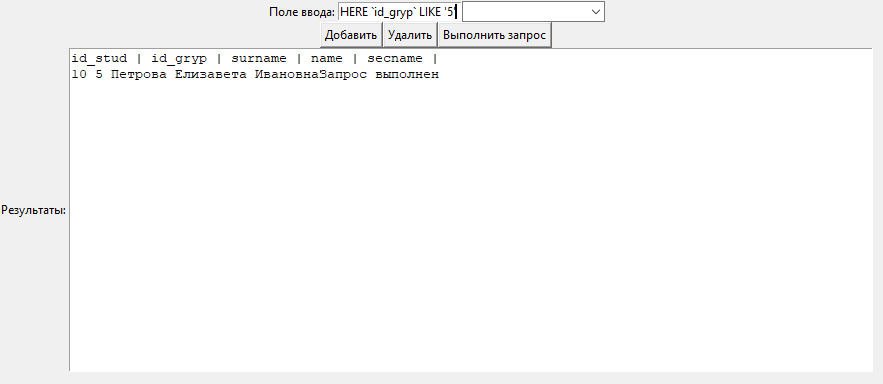


Рисунок 6.6 – функция выполнения SQL запросов

# 7 Запуск одноузлового Kubernetes-кластера с помощью Minikube



Рисунок 7.1 - установка Kubectl: часть 1

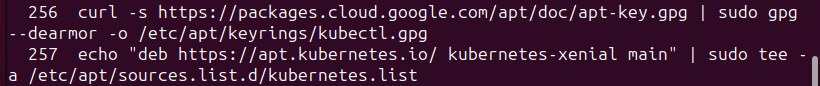


Рисунок 7.2- установка Kubectl: часть 2



Рисунок 7.3- установка Kubectl: часть 3

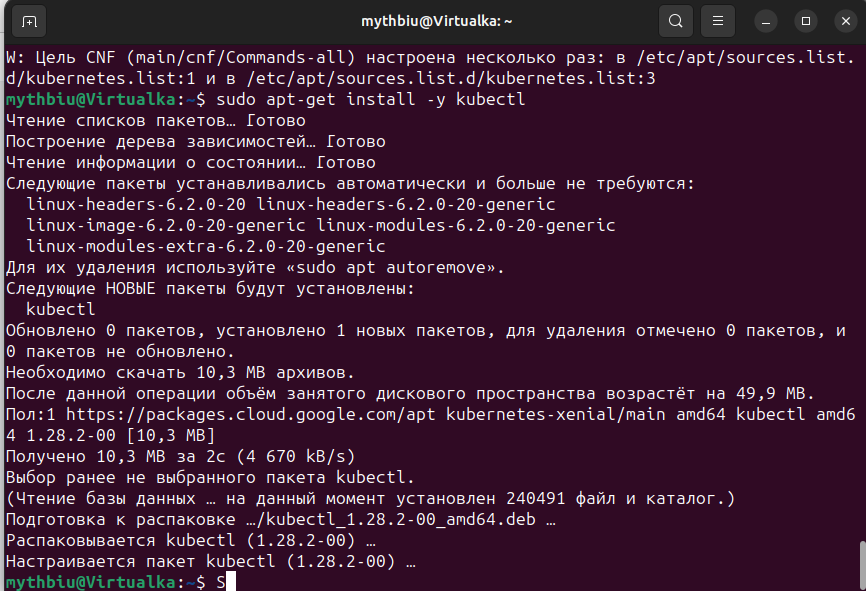


Рисунок 7.4- установка Kubectl: часть 4

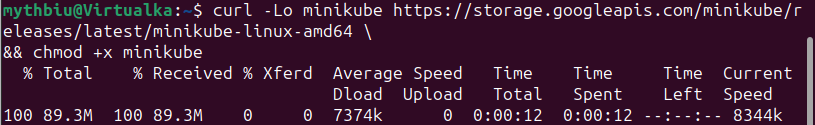


Рисунок 7.5- установка Minikube: часть 1

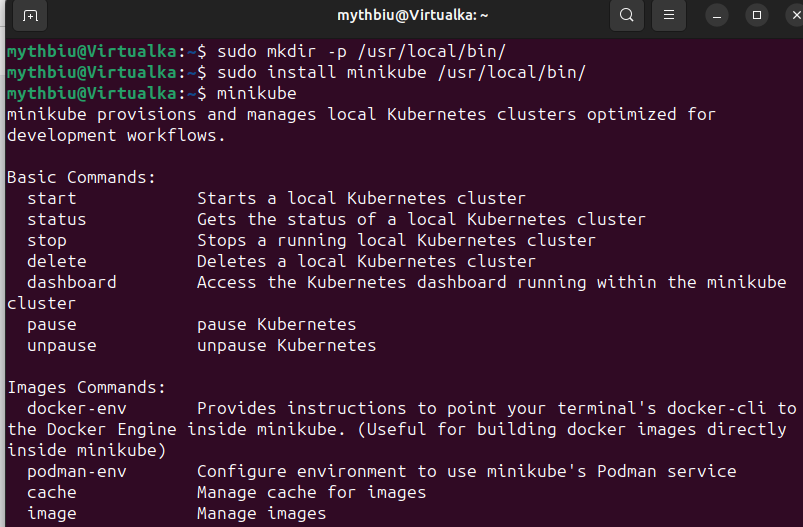


Рисунок 7.6- установка Minikube: часть 2

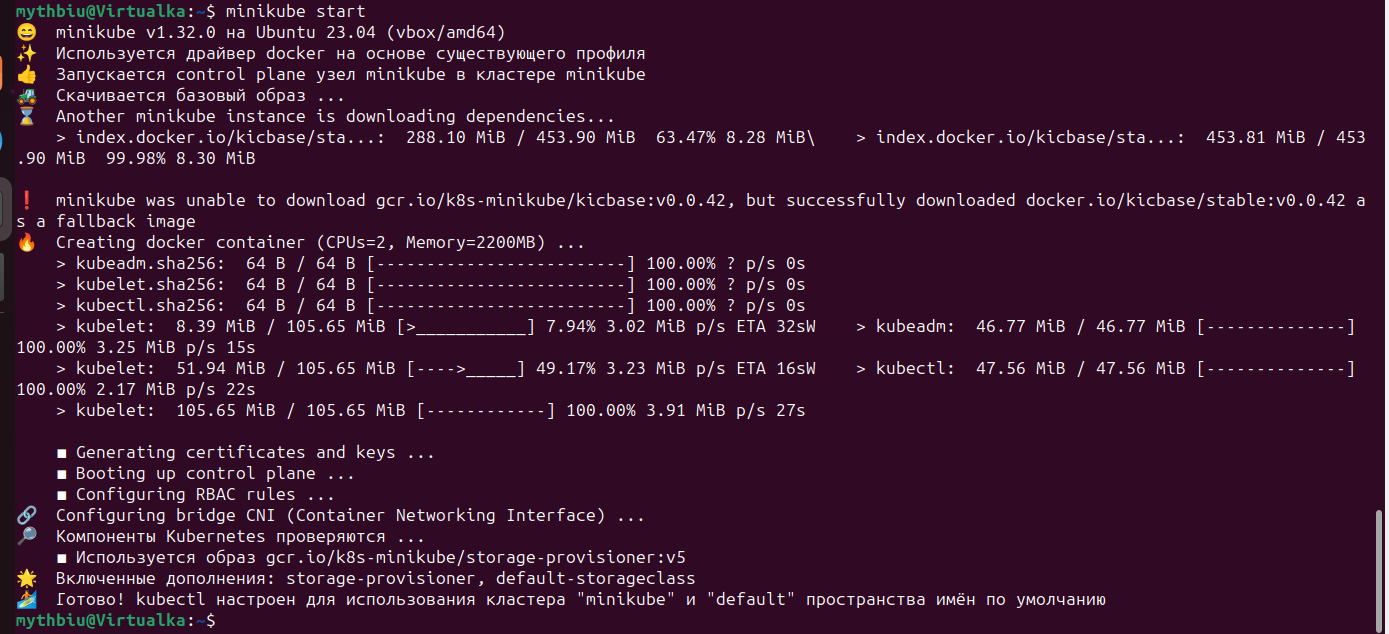


Рисунок 7.7- запуск Minikube

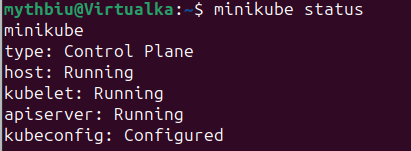


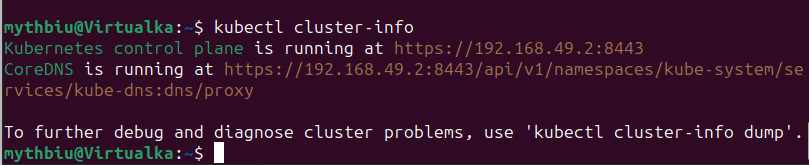
Рисунок 7.8- статус кластера: часть 1

Рисунок 7.9- статус кластера: часть 2

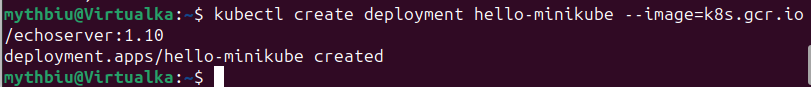


Рисунок 7.11- создание развёртывания:

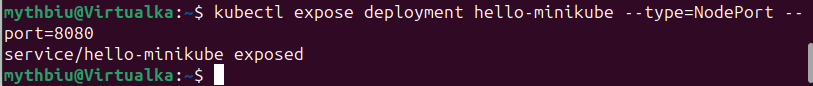


Рисунок 7.12- создание сервиса

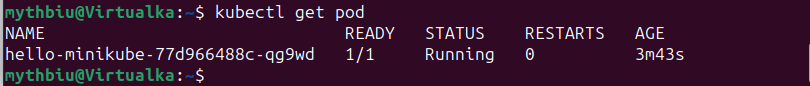


Рисунок 7.13- статус пода

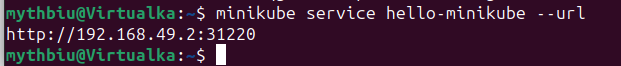


Рисунок 7.14- URL сервиса

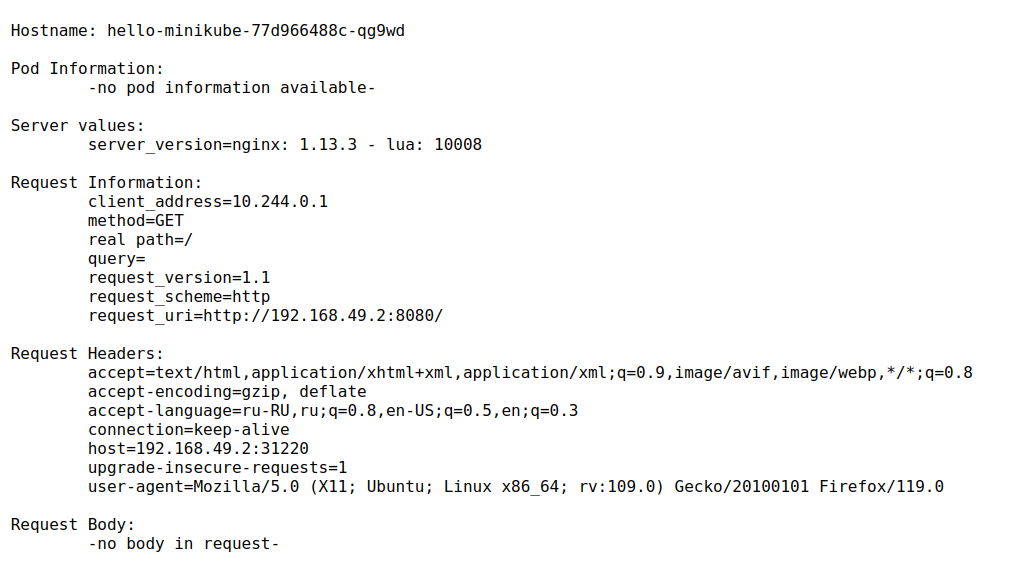


Рисунок 7.15- результат ввода URL в браузер

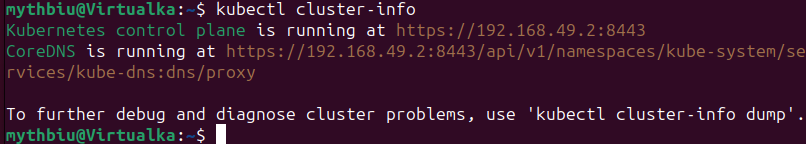


Рисунок 7.16- состояние кластера

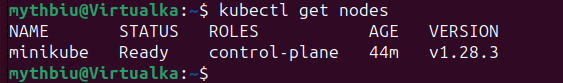


Рисунок 7.17- информация о нодах

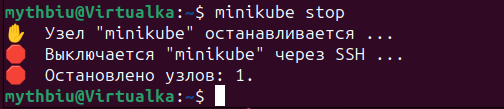


Рисунок 7.18- Остановка кластера

8 Запуск кластера с помощью Kubeadm

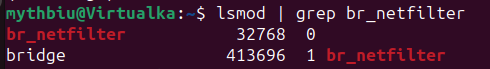


Рисунок 8.1- проверка нетфильтра



Рисунок 8.2- отключение свапа



Рисунок 8.3- Установка Kubeadm и Kubelet: часть 1



Рисунок 8.4- Установка Kubeadm и Kubelet: часть 2

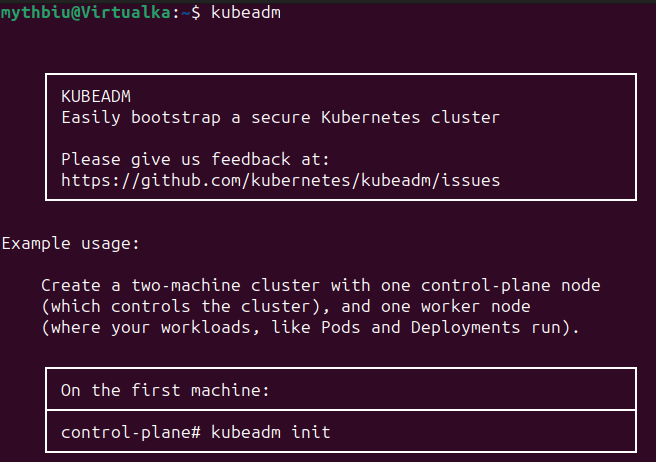


Рисунок 8.5– проверка установки Kubeadm



Рисунок 8.6– проверка установки Kubelet

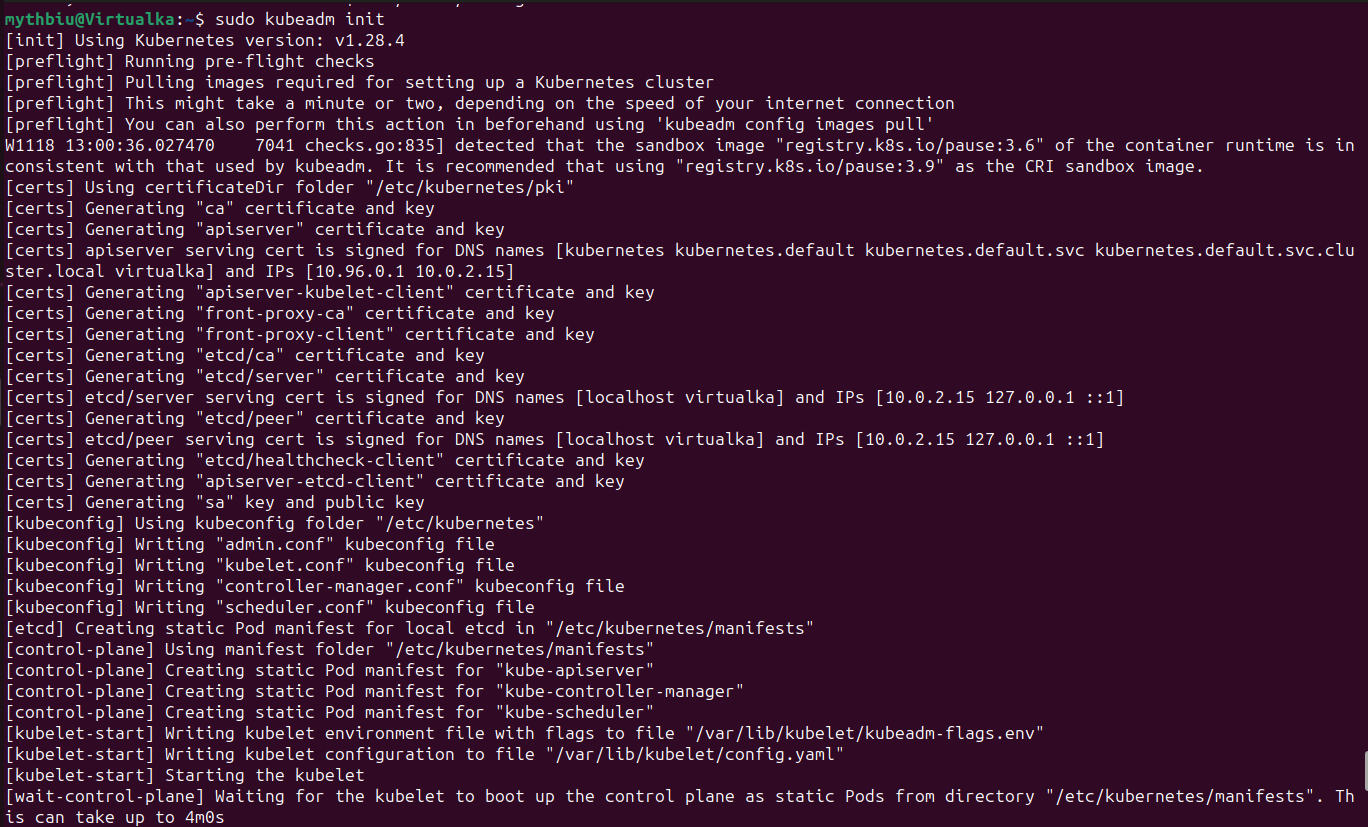


Рисунок 8.7– инициализация узла уровня управления: часть 1

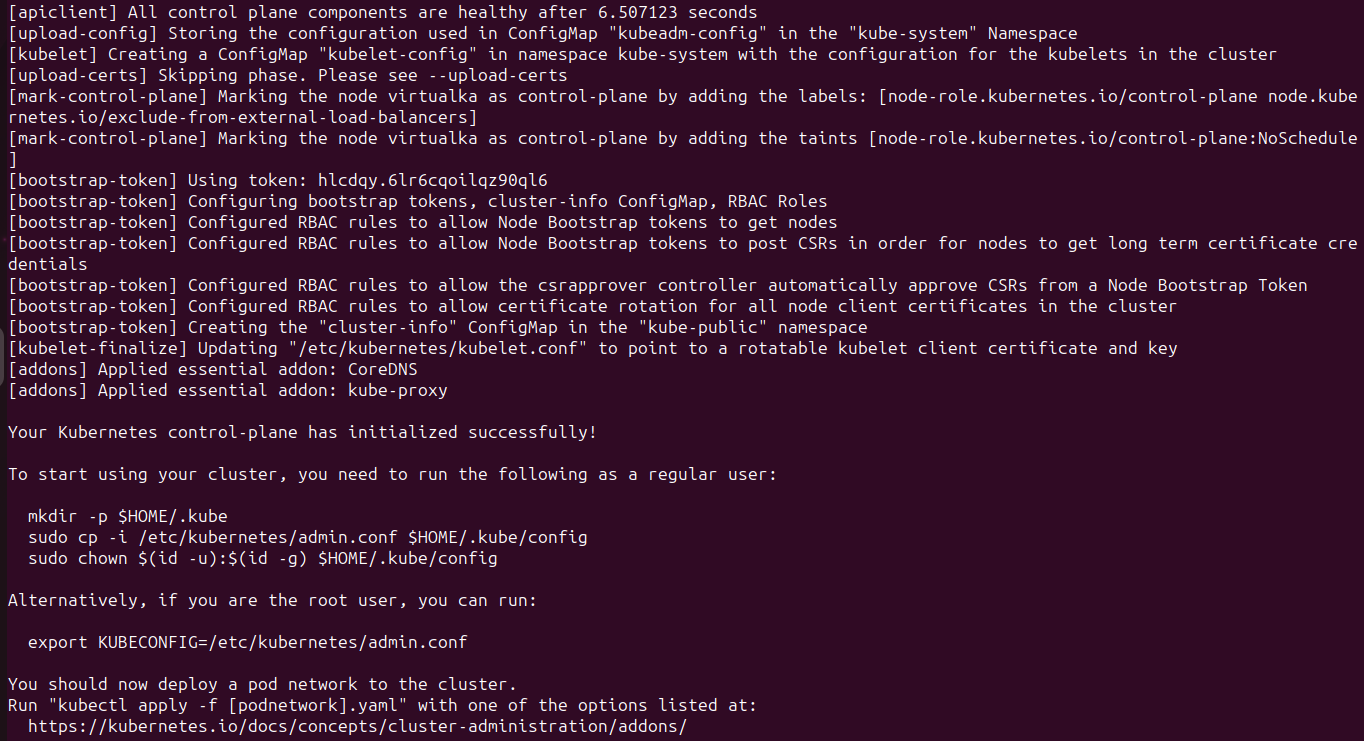


Рисунок 8.9– инициализация узла уровня управления: часть 2

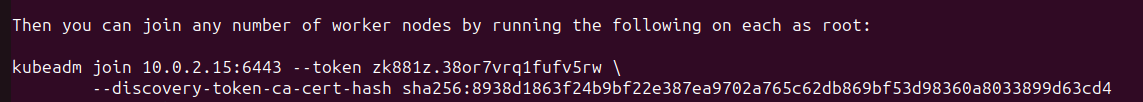


Рисунок 8.10– инициализация узла уровня управления: часть 3

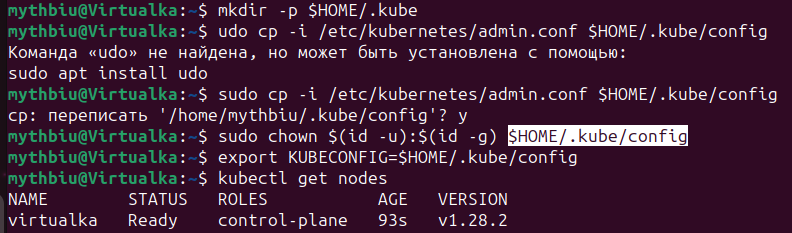


Рисунок 8.11- создание конфига для Kubectl:

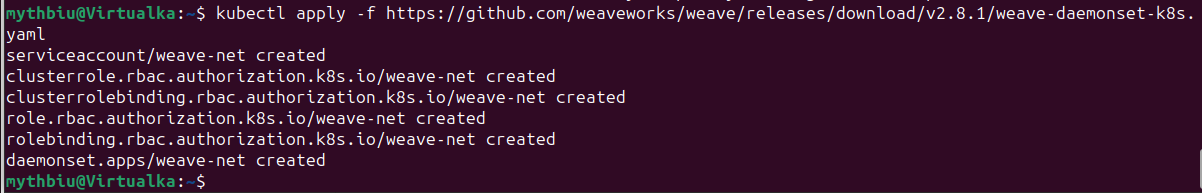


Рисунок 8.12– установка сетевой надстройки Pod

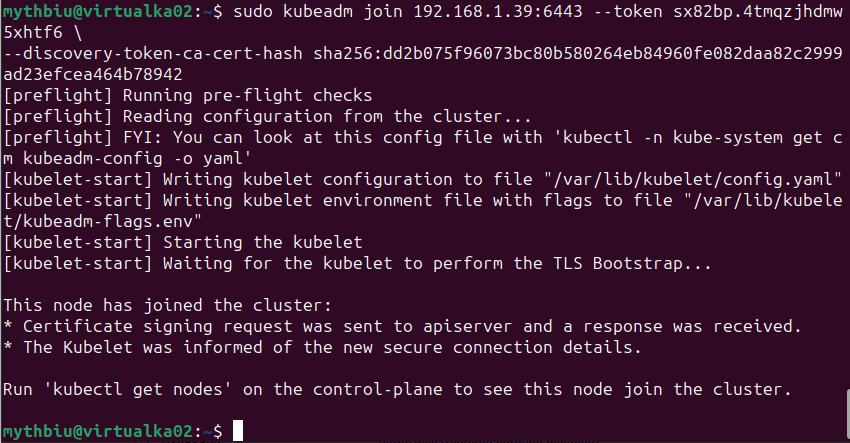


Рисунок 8.13- подключение другой машины к узлу

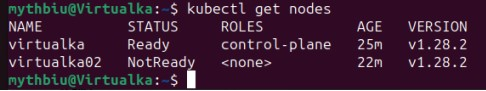


Рисунок 8.14- проверка кластера на узле управления

# Заключение

Данная курсовая работа состоит из вводной, теоретической и практической частей. Были рассмотрены различные программы для создания программного продукта для работы с базой данных.

Разработана информационная система для учебной части колледжа. Уровень знаний в области информационных технологий повысился и были приобретены навыки по выбору платформ и программных сред разработки информационных систем.

Для разработки информационной системы были выбраны веб-интерфейс для администрирования СУБД MySQL и языка программирования – Python. Были изучены основы принципов работы реляционной базы данных, языка запросов SQL и интеграции базы данных с Python.

Также в ходе работы были получены дополнительные знания и навыки работы с проектом для автоматизации развертывания приложений в виде переносимых автономных контейнеров Docker и с проектом для развёртывания контейнеров Kubernetes. Был запущен одноузловой и многоузловый кластер.

Разработанная информационная система позволит упростить работу с информацией и облегчит работу персонала колледжа.

# Список использованных источников

1. «Жизненный цикл программного обеспечения» [Электронный ресурс] Режим доступа: https://qaevolution.ru/zhiznennyj-cikl-programmnogo-obespecheniya/
2. Документация по PyMySQL [Электронный ресурс] Режим доступа: https://pymysql.readthedocs.io/en/latest/index.html
3. Хабр, статья: «Open Server — профессиональный инструмент веб-разработчика под Windows» [Электронный ресурс] Режим доступа: https://habr.com/ru/articles/137388/
4. Oracle, статья: «Что такое база данных?» [Электронный ресурс] Режим доступа: https://www.oracle.com/cis/database/what-is-database/
5. Skillfactory, статья: «Python: что это за язык программирования и где применяется» [Электронный ресурс] Режим доступа: https://blog.skillfactory.ru/glossary/python/
6. Yandex, статья: «Как устроен язык SQL и почему он так востребован» [Электронный ресурс] Режим доступа: https://practicum.yandex.ru/blog/chto-takoe-sql/
7. Tkinter в Python [Электронный ресурс] – Режим доступа: https://skillbox.ru/media/code/pishem-desktopprilozhenie-na-python-s-pomoshchyu-tkinter/
8. Основы работы в phpMyAdmin [Электронный ресурс] – Режим доступа: https://timeweb.com/ru/docs/virtualnyj-hosting/bazy-dannyh/osnovy-raboty-v-phpmyadmin/

# Приложение А - Техническое задание

**Техническое задание**

**«УТВЕРЖДАЮ»**

« » 2022 г.

Техническое задание

На разработку информационной системы для учебной части колледжа

«СамПГУ»

Листов 3

**Разработал Часовских Г.С.**

Самара 2023

**A.1 Введение**

Настоящее техническое задание распространяется на разработку информационной системы для использования сотрудниками колледжа в целях облегчения работы с информацией.

**A.2 Основание для разработки**

A.2.1. Программа разрабатывается на основе заказа от колледжа.

A.2.2. Наименование работы:

“Создание информационной системы для учебной часть колледжа «СамПГУ»”.

A.2.3. Исполнитель: **Часовских Г.С**.

A.2.4. Соисполнители: нет.

**A.3 Назначение**

Программа предназначена для работы с базой данных колледжа, вносить новую информацию, удалять ненужную и делать выборку необходимой информации по какому-либо условию.

**A.4 Требования к программе или программному изделию**

A.4.1 Требования к функциональным характеристикам

Программа должна обеспечивать возможность выполнения следующих функций:

* + - ввод новых данных в базу данных;
    - хранение информации в базе данных;
    - выбор нужной информации из базы данных;
    - удаление ненужной информации.

Организация входных и выходных данных. Входные данные поступают с клавиатуры.

Выходные данные отображаются на экране и при необходимости выводятся на печать.

A.4.2 Требования к надежности

Предусмотреть контроль вводимой информации. Вводимая в базу данных информация должна соответствовать реальным данным.

A.4.3 Требования к составу и параметрам технических средств

Система должна работать на IBM-совместимых персональных компьютерах.

Минимальная конфигурация:

* + - процессор 4 ядра (8 логических потоков), частота – 3 ГГц и больше;
    - объем оперативного запоминающего устройства 4 Гб и больше;
    - объем свободного места на жестком диске 10 Гб (зависит от количества хранимых данных).

Рекомендуемая конфигурация:

* + - процессор 4 ядра (8 логических потоков), частота – 3,5-4 ГГц и больше;
    - объем оперативного запоминающего устройства 8 Гб и больше;
* объем свободного места на жестком диске 50 ГБ (зависит от количества хранимых данных).

A.4.4 Требования к программной совместимости:

Программа должна работать под управлением операционных систем РЕД ОС и Astra Linux.

A.4.5 Требования к простоте эксплуатации:

Полное изучение функционала программы персоналом не должно превышать 20 минут. Поставляемая с программой инструкция должна объяснять весь функционал программы.

A.4.6 Требования к безопасности:

Программа должна иметь встроенную систему безопасности с использованием проверки подключения к серверу базы данных.

A.4.7 Требования к интерфейсу:

Интерфейс программы должен быть интуитивно понятным. В интерфейсе должен преобладать белый цвет. Программа должна быть выполнена на русском языке.

A.4.8 Требования к производительности:

Время отклика программы не должно превышать 3 секунд. Недопустимо зависание программы в состоянии «Не отвечает».

**A.5 Требования к программной документации:**

A.5.1 Разрабатываемая программа должна включать справочную информацию о работе программы, описания методов сортировки и подсказки учащимся.

A.5.2 В состав сопровождающей документации должны входить:

* пояснительная записка, содержащая описание разработки;
* руководство пользователя.

**A.6 Стадии и этапы разработки**

|  |  |  |  |
| --- | --- | --- | --- |
| №  этапа | Название этапа | Сроки этапа | Чем заканчивается этап |
| 1 | Изучение предметной области. Проектирование системы. Разработка предложений по реализации  системы | 08.09.2023-  30.09.2023 | Предложения по работе системы. Акт начала работы. |
| 2 | Разработка информационной системы для учебной части колледжа | 02.10.2023-  30.10.2023 | Информационная система, решающая поставленные задачи. Акт приемки. |
| 3 | Тестирование и отладка системы. Внедрение системы во всей системе колледжа | 03.11.2023-  30.11.2023 | Готовая информационная система для колледжа. Программная документация.  Акт сдачи. |

**A.7 Порядок контроля и приемки**

После передачи Исполнителем отдельного функционального модуля программы Заказчику последний имеет право тестировать модуль в течение 7 дней. После тестирования заказчик должен принять работу по данному этапу или в письменном виде изложить причину отказа принятия. В случае обоснованного отказа Исполнитель обязуется доработать модуль.

# Приложение B - Эскизный проект

**Эскизный проект**

Утверждаю

Руководитель (заказчика ИС)

Личная подпись\_\_\_\_\_\_\_\_\_\_\_\_\_

Расшифровка подписи\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Печать

Дата «\_\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_202\_

Утверждаю

Руководитель (разработчика ИС)

Личная подпись\_\_\_\_\_\_\_\_\_\_\_\_\_

Расшифровка подписи\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Печать

Дата «\_\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_202\_

Эскизный проект на создание информационной системы

Система управления базой данных

наименование вида ИС

Колледж «СамПГУ»

наименование объекта информатизации

СУБД «СамПГУ»

краткое наименование ИС

На 5 листах

Действует с 1.09.2023

**Ведомость эскизного проекта**

На предыдущих стадиях разработки СУБД «СамПГУ» были составлены и утверждены следующие документы:

* Техническое задание на создание информационной системы СУБД «СамПГУ», разработанное на основании ГОСТ 34.602—89 на написание ТЗ на автоматизированные системы управления от 10.09.2023 г.

**Пояснительная записка к эскизному проекту.**

**Общие положения.**

Данный документ является эскизным проектом на создание Системы Управления Базой Данных для учебной части колледжа Российской Федерации (СУБД «СамПГУ»).

Перечень организаций, участвующих в разработке системы, на стадии разработки, а также ее цели и назначение указаны в техническом задании на создание информационной системы.

**Решения по структуре системы:**

СУБД «СамПГУ» будет представлять собой персональную систему управления локальной базой данных, работающей на нескольких компьютерах.

Система будет управлять реляционной базой данных, представляющей собой набор связанных между собой таблиц, доступ к которым осуществляется с помощью ключей или индексов. Сведения в одной таблице могут отражать сведения из другой, и при изменении сведений в первой таблице эти изменения немедленно отображаются во второй. Таким образом будет достигнута непротиворечивость данных.

Общая структура базы данных:

1. Дисциплины:

* Код дисциплины
* Код группы
* Код преподавателя
* Название
* Приказ
* Год создания

1. Группы:

* Код группы
* Код специальности
* Название
* Тип группы
* Курс

1. Кафедры:

* Код кафедры
* Название
* Глава кафедры
* Приказ
* Год создания
* Адрес
* Телефон

1. Специальности:

* Код специальность
* Название
* Год создания

1. Студенты:

* Код студента
* Код группы
* Имя
* Фамилия
* Отчество

1. Личные дела студентов:

* Код студента
* Год поступления
* Приказ о поступлении
* Стипендия
* Дата рождения
* Адрес
* Электронная почта
* Телефон

1. Преподаватели:

* Код преподавателя
* Код кафедры
* Имя
* Фамилия
* Отчество

1. Личные дела преподавателей:

* Код преподавателя
* Год найма
* Адрес
* Электронная почта
* Телефон
* Заработная плата
* Степени
* Год преподавания
* Дата рождения

**Решение по режимам функционирования и работы системы:**

СУБД «СамПГУ» будет функционировать и однопользовательском режиме, а также будет способна:

* просматривать записи базы данных (и том числе и при помощи фильтров);
* добавлять новые записи;
* удалять записи.

**Состав функций комплексов задач, реализуемых системой:**

Автоматизированная система должна выполнять следующие функции:

* сделать запись о студенте;
* удалить информацию о студенте;
* сделать запись о преподаватели;
* удалить информацию о преподаватели;
* сделать запись о кафедре;
* удалить запись о кафедре;
* сделать запись о специальность;
* удалить запись о специальности;
* сделать запись о группе;
* удалить запись о группе;
* сделать выборку информации из нужной таблицы.

**Решения по составу программных средств, языками деятельности, алгоритмам процедур и операций и методам их реализации:**

Для реализации АС будет использоваться среда программирования PyCharm и язык программирования Python. Для работы с базой данных будет использован язык MySQL и библиотеки для языка программирования Python, предназначенные для работы с MySQL.

Функциональные диаграммы:

Изображение выглядит как снимок экрана, текст, диаграмма, линия

Автоматически созданное описание

Рисунок B.1 – функциональная IDEF0 схема

Составил

Должность исполнителя Главный разработчик

ФИО исполнителя Часовских Г.С.

Подпись \_\_\_\_\_\_\_\_\_\_\_\_

Дата 10.09.2023

# Приложение C - Технический проект

|  |  |  |
| --- | --- | --- |
| УТВЕРЖДАЮ |  | УТВЕРЖДАЮ |
|  |  |  |
|  |  |  |
|  |  | \_\_\_\_\_\_\_\_\_\_\_\_ |
| Подпись ФИО |  | Подпись ФИО |
| «\_\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_2023 г. |  | «\_\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_2023 г. |

СУБД «СамПГУ»

Пояснительная записка к техническому проекту

C.1 Аннотация

Настоящий документ содержит Пояснительную записку к Техническому проекту «СУБД «СамПГУ»», в которой приведено краткое изложение содержания проекта и приводятся основные сведения об Информационной Системе.

**C.2 Общие положения**

**C.2.1 Сведения о проекте**

|  |  |
| --- | --- |
| **Заказчик**: | Колледж «СамПГУ». |
| **Исполнитель**: | Часовских Г.С. |
| **Соисполнители**: | Отсутствуют. |

**C.2.2 Основание для разработки**

Работа выполняется на основании заказа от Колледжа «СамПГУ».

**C.2.3 Цели, назначение и область использования системы**

Целью работ является повышение качества, эффективности учёта информации, а также облегчение работы с информацией и минимизация затрат времени на подобные работы.

**C.2.4 Перечень нормативных документов, используемых при проектировании системы**

При проектировании ИС использовались следующие документы и материалы:

* ГОСТ 34.201-89. «Виды, комплектность и обозначение документов при создании автоматизированных систем».
* ГОСТ 34.601-90. «Автоматизированные системы. Стадии создания».
* РД 50-34.698-90. «Автоматизированные системы. Требования к содержанию документов».

**C.3 Описание процесса деятельности**

**C.3.1 Виды автоматизируемой деятельности**

В рамках Информационной системы колледжа «СамПГУ» автоматизируются следующие виды деятельности:

* Ввод информации о студентах, преподавателях, кафедрах, дисциплинах и групп в базу данных колледжа.
* Выборка необходимой информации из базы данных.
* Удаление ненужной (устаревшей) информации из базы данных.

В рамках ввода информации в базу данных происходит заполнение базы данных для дальнейшей работы.

В рамках выборки необходимой информации пользователь получает данные, заданные по необходимому критерию определённой задачи.

В рамках удаления ненужной информации идёт очистка базы данных от ненужной и устаревшей информации для предотвращения ошибок в расписаниях и отчётах.

**C.4 Основные технические решения**

**C.4.1 Средства разработки и состав программного обеспечения системы**

При проектировании и разработке программного обеспечения были использованы архитектурные решения и программные продукты, удовлетворяющие требованиям заказчика.

Программное обеспечение создавалось на модульной основе с использованием объектно-ориентированного подхода, позволяющего добавлять/изменять функциональные возможности подсистемы.

В качестве средства разработки программного обеспечения будет использоваться среда программирования PyCharm и язык программирования Python. Для работы с базой данных будет использован язык MySQL и библиотеки для языка программирования Python, предназначенные для работы с MySQL.

**C.5 Дополнительные материалы**

Изображение выглядит как диаграмма, линия, текст, Параллельный

Автоматически созданное описание

Рисунок C.1 - Функциональная схема

Изображение выглядит как текст, диаграмма, План, Технический чертеж

Автоматически созданное описание

Рисунок C.2 - Дерево диалога