ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ И НАУКИ ГОРОДА МОСКВЫ

Государственное бюджетное профессиональное образовательное учреждение города Москвы

«Московский государственный колледж электромеханики и информационных технологий»

**Практическая работа No3. Формирование требований к базе данных.**

[УП.11 Разработка, администрирование и защита баз данных](https://classroom.google.com/c/NDI2NDY2OTEwNTAw)

Выполнил студент группы: 3ИП-11-19

Терехин Михаил Александрович

Преподаватель: Басыров Сергей Амирович

**Содержание**

[ВВЕДЕНИЕ 3](#_Toc88478525)

[ОСНОВНАЯ ЧАСТЬ 4](#_Toc88478526)

[1 Обосновать выбор СУБД. 4](#_Toc88478527)

[2 Описать основные сущности и их характеристики, которые будут храниться в БД (в каком виде и как). 4](#_Toc88478528)

[3 Описать основные сценарии использования СУБД с помощью use-case. 4](#_Toc88478529)

[4 Составить список API методов, предоставляемой БД для работы «из вне». 4](#_Toc88478530)

[5 Определить список пользователей их роли. 4](#_Toc88478531)

[ЗАКЛЮЧЕНИЕ 5](#_Toc88478532)

[СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ 6](#_Toc88478533)

# ВВЕДЕНИЕ

Цель – Научиться формировать требования к базе данных

(программному продукту).

Актуальность – моя работа сделана для улучшенного представления о датчиках и работы.

Задачи:

1) Обосновать выбор СУБД;

2) Описать основные сущности и их характеристики, которые будут

храниться в БД (в каком виде и как);

3) Описать основные сценарии использования СУБД с помощью use-

case;

4) Составить список API методов, предоставляемой БД для работы «из

вне»;

5) Определить список пользователей их роли.

Предмет исследования – СУБД.

Объект исследования – БД датчики учета.

# ОСНОВНАЯ ЧАСТЬ

# 1 Обосновать выбор СУБД.

Сравним СУБД друг-другом, чтобы стало понятно.

MySQL – одна из самых популярных баз данных для веб-приложений. Фактически, является стандартом de facto для веб-серверов, которые работают под управлением операционной системы Linx. MySQL - это бесплатный пакет программ, однако новые версии выходят постоянно, расширяя функционал и улучшая безопасность. Существуют специальные платные версии, предназначенные для коммерческого использования. В бесплатной версии наибольший упор делается на скорость и надежность, а не на полноту функционала, который может стать и достоинством и недостатком - в зависимости от области внедрения.

**Достоинства**

* Распространяется бесплатно
* Прекрасно документирована
* Предлагает много функций, даже в бесплатной версии
* Пакет MySQL включен в стандартные репозитории наиболее распространённых дистрибутивов операционной системы Linux, что позволяет устанавливать её элементарно
* Поддерживает набор пользовательских интерфейсов
* Может работать с другими базами данных, включая DB2 и Oracle.

**Недостатки**

* Придётся потратить много времени и усилий, чтобы заставить MySQL выполнять несложные задачи, хотя другие системы делают это автоматически, например: создавать инкрементные резервные копии.
* Отсутствует встроенная поддержка XML или OLAP.
* Для бесплатной версии доступна только платная поддержка.

Microsoft SQL-сервер – это система управления базами данных, движок которой работает на облачных серверах, а также локальных серверах, причем можно комбинировать типы применяемых серверов одновременно. Вскоре после выпуска Microsoft SQL сервер 2016, Microsoft адаптировала продукт для операционной системы Linux, а на Windows-платформе он работал изначально.

**Достоинства**

* Продукт очень прост в использовании
* Текущая версия работает быстро и стабильно
* Движок предоставляет возможность регулировать и отслеживать уровни производительности, которые помогают снизить использование ресурсов.
* Вы сможете получить доступ к визуализации на мобильных устройствах.
* Он очень хорошо взаимодействует с другими продуктами Microsoft.

**Недостатки**

* Цена для юридических лиц оказывается неприемлемой для большей части организаций.
* Даже при тщательной настройке производительности корпорация SQL Server способен занять все доступные ресурсы.
* Сообщается о проблемах с использованием службы интеграции для импорта файлов.

MongoDB — документно-ориентированная система управления базами данных, не требующая описания схемы таблиц. Система поддерживает ad-hoc -запросы: они могут возвращать конкретные поля документов и пользовательские JavaScript -функции. Поддерживается поиск по регулярным выражениям. Также можно настроить запрос на возвращение случайного набора результатов.

**Достоинства**

* Скорость и простота в использовании
* Движок поддерживает json и другие традиционные документы NoSQL.
* Данные любой структуры могут быть сохранены/прочитаны быстро и легко.

**Недостатки**

* SQL не используется в качестве языка запросов.
* Инструменты для перевода SQL-запросов в MongoDB доступны, но их следует рассматривать именно как дополнение.
* Программа установки может занять много времени.

Вывод:

MySQL – Идеально подходит для: организаций, которым требуется надежный инструмент управления базами данных, но бесплатный.

Microsoft SQL-сервер – Идеально подходит для: крупных организаций, которые уже используют ряд продуктов Microsoft.

MongoDB – Подходит для организаций, работающих с разнородными данными, которые тяжело поддаются классификации. Для внедрения потребуются высококлассные специалисты.

# 2 Описать основные сущности и их характеристики, которые будут храниться в БД (в каком виде и как).

Рисунок 1 – Диаграмма сущности.

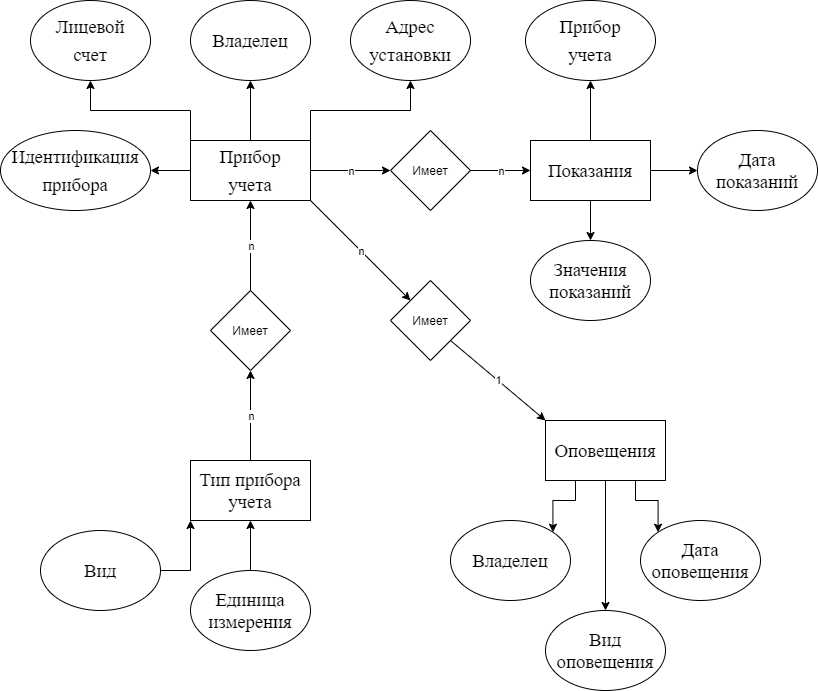


Рисунок 1– Диаграмма сущности.

# 3 Описать основные сценарии использования СУБД с помощью use-case.

UML диаграмма использования, иначе use-cases составляется для того, чтобы наглядно представить функциональные возможности разрабатываемой системы. Основными элементами этой диаграммы являются:

1) акторы;

2) ситуации;

3) стрелки.

Акторы описывают собой непосредственно участников, взаимодействующие с системой. Ситуации описывают собой действия или процессы, допустимые в системе.

Стрелки связывают между собой акторов и ситуации. В зависимости от её визуального вида, она несёт дополнительные сведения о связи между актором и ситуацией.

Рисунок 2 – use-case диаграмма.

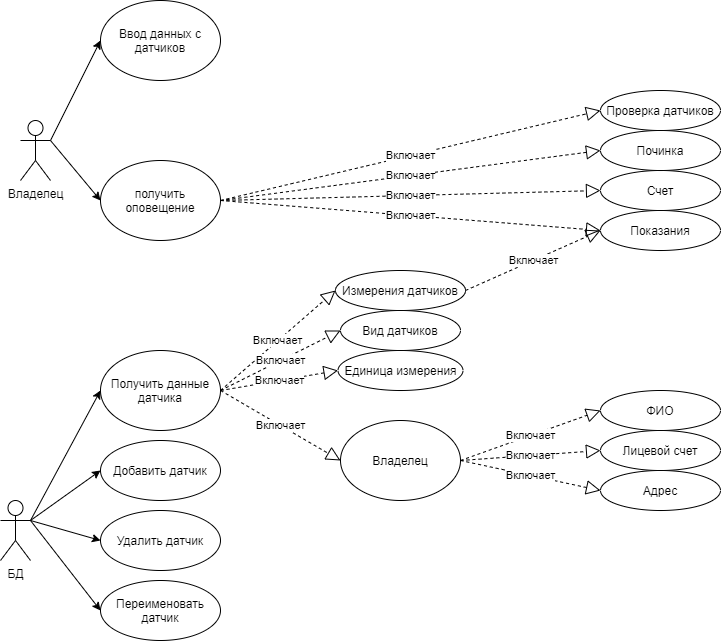


Рисунок 2 - use-case диаграмма.

# 4 Составить список API методов, предоставляемой БД для работы «из вне».

Список команд представлен в Таблице 1 – список аргументов.

Таблица 1 – список аргументов.

|  |  |
| --- | --- |
| **Метод** | **Аргументы** |
| Добавить прибор учёта | Идентификация прибора (текст),  Лицевой счет (число). Владелец (текст). Адрес установки (текст), Тип прибора учета(текст (связь с Вид)). |
| Удалить прибор учёта | Идентификатор прибора |
| Изменить лицевой счёт | Идентификатор прибора, новый лицевой счёт |
| Изменить владелец | Идентификатор прибора, владелец. |
| Изменить адрес установки | Идентификатор прибора, адрес установки. |
| Посмотреть все данные по приборам учета. | Идентификатор прибора. |
| Поиск датчика | Идентификатор прибора, владелец. |
| Добавить тип прибора учета. | Вид (текст), единица измерения (текст). |
| Удалить тип прибора учёта. | Идентификатор тип прибора учета. |
| Изменить вид. | Идентификатор тип прибора учета, вид. |
| Изменить единица измерения. | Идентификатор тип прибора учета, единица измерения. |
| Посмотреть все данные по типам прибора учёта. | Идентификатор тип прибора учета. |
| Поиск типа прибора учёта. | Идентификатор тип прибора учета, вид. |
| Добавить оповещения. | Владелец (текст (связь с Владелец), Дата оповещения (дата), Вид оповещения (текст). |
| Удалить оповещения | Идентификатор оповещения. |
| Изменить дата оповещения. | Идентификатор оповещения, дата оповещения. |
| Изменить вид оповещения. | Идентификатор оповещения, вид оповещения. |
| Посмотреть все данные по оповещению. | Идентификатор оповещения. |
| Поиск вида оповещения. | Идентификатор оповещения, вид оповещения |
| Добавить показания. | Прибор учёта (текст (связь с Идентификация прибора)), Дата показаний(дата), Значение показаний (число). |
| Удалить показания. | Идентификатор показания. |
| Изменить дата показания. | Идентификатор показания, дата показания. |
| Изменить значение показаний. | Идентификатор показания, значение показания. |
| Посмотреть все данные по показаниям. | Идентификатор показания. |
| Поиск показания. | Идентификатор показания, дата показания. |

# 5 Определить список пользователей их роли.

Включает 3 роли:

Пользователя – позволяет узнать состояние счетчиков и вводить данные с показателей.

Администратор – ответственное лицо за функционирование информационной системы; он управляет режимом использования данных.

Сотрудник – ответственного за пользователей базы данных. Этот сотрудник получит возможность создавать, изменять и удалять объекты пользователей баз данных, а также создавать схемы. Других прав в базе данных у него нет.

# ЗАКЛЮЧЕНИЕ

Впроцессе выполнения практической работы было сформировано требоване к базе данных. В результате выполнения работы сформировалось представление о дальнейшей разработке сервиса. Поставленные задачи были выполнены в ходе работы.

# СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1) https://app.diagrams.net/

2) https://evergreens.com.ua/ru/articles/uml-diagrams.html

3) https://tproger.ru/translations/sql-recap/