

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра МО ЭВМ**

**ИНДИВИДУАЛЬНОЕ ДОМАШНЕЕ ЗАДАНИЕ**  
**по дисциплине «Введение в нереляционные базы данных»**  
**Тема: Сервис хранения экспериментов инструмента pybullet**

Студенты гр. 0304

\_\_\_\_\_

Докучаев Р.А.

\_\_\_\_\_

Никитин Д.Э.

\_\_\_\_\_

Свечников И.В.

Преподаватель

\_\_\_\_\_

Заславский М.М.

Санкт-Петербург

2023

## **ЗАДАНИЕ НА КУРСОВУЮ РАБОТУ (КУРСОВОЙ ПРОЕКТ)**

Студенты Иванов И.И.

Докучаев Р.А.

Никитин Д.Э.

Свечников И.В.

Группа 0304

Тема проекта: Сервис хранения экспериментов инструмента pybulletdrones

Исходные данные:

Необходимо создать простое веб-приложение для импорта/хранения/поиска/визуализации результатов экспериментов в эмуляторе pybulletdrones

Содержание пояснительной записки:

«Содержание»

«Введение»

«Сценарии использования»

«Модель данных»

«Разработанное приложение»

«Выводы»

Предполагаемый объем пояснительной записки:

Не менее 20 страниц.

Дата выдачи задания: 26.09.2023

Дата сдачи реферата: 25.12.2023

Дата защиты реферата: 25.12.2023

Студенты гр. 0304

\_\_\_\_\_

Докучаев Р.А.

\_\_\_\_\_

Никитин Д.Э.

\_\_\_\_\_

Свечников И.В.

Преподаватель

\_\_\_\_\_

Заславский М.М.

## **АННОТАЦИЯ**

В качестве индивидуального домашнего задания была выбрана тема «Сервис хранения экспериментов инструмента pybulletdrones», которая предполагает разработку простого веб-приложения, осуществляющего поддержку работы с результатами экспериментов в эмуляторе. Исходный код разработанного приложения можно найти по ссылке: <https://github.com/moevm/nosql2h23-drones>.

## **SUMMARY**

The topic “Experiment storage service for the pybulletdrones tool” was chosen as an individual homework assignment, which involves the development of a simple web application that supports working with the results of experiments in the emulator. The source code of the developed application can be found at the link: <https://github.com/moevm/nosql2h23-drones>.

## СОДЕРЖАНИЕ

	Введение	6
1.	Сценарии использования	7
1.1.	Макет UI	7
1.2.	Сценарии использования для задач	7
1.2.1.	Добавление/удаление результатов проведённого эксперимента в БД	7
1.2.2.	«Экспорт/импорт» результатов проведённого эксперимента в БД	8
1.2.3.	Просмотр графических результатов проведённого эксперимента	9
1.2.4.	Поиск эксперимента среди проведённых и занесённых в БД	10
1.2.5.	Просмотр таблицы с данными по проведённому эксперименту	10
2.	Модель данных	12
2.1.	Нереляционная модель данных	12
2.2.	Реляционная модель данных	16
2.3.	Сравнение моделей данных	20
3.	Разработанное приложение	22
3.1.	Краткое описание	22
3.2.	Снимки экрана приложения	22
	Заключение	25

## **ВВЕДЕНИЕ**

Цель работы – разработать веб-приложение, которое будет хранить результаты экспериментов в эмуляторе в удобном для пользователя электронном виде с возможностями дополнения новыми данными, удаления утративших актуальность данных, поиска и фильтрации данных.

Также были выработаны следующие качественные требования к решению: необходимо разработать приложение с использованием СУБД – MongoDB, в которой будут храниться данные экспериментов в эмуляторе pybullet, с возможностью локального разворота приложения с помощью docker.

# 1. СЦЕНАРИИ ИСПОЛЬЗОВАНИЯ

## 1.1. Макет UI

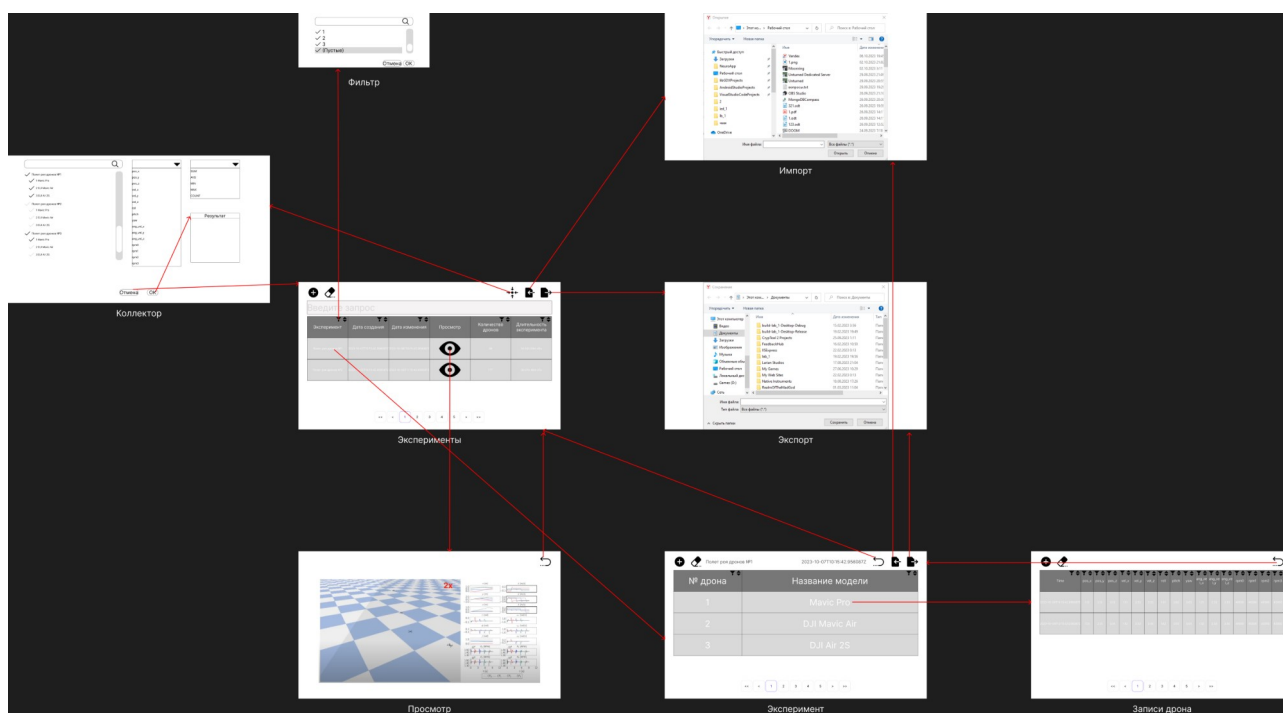


Рис. 1 – Макет UI

## 1.2. Сценарии использования

### 1.2.1. Добавление/удаление результатов проведённого эксперимента

в БД

*Основной сценарий:*

1. Система выводит на экран веб-страницу со списком уже записанных экспериментов в виде таблицы
  2. Пользователь нажимает на кнопку “добавление”/“удаление”
  3. Система открывает во всплывающем окне форму для записи эксперимента в таблицу либо список экспериментов с возможностью отметить эксперименты к удалению
  4. Пользователь заносит данные либо выбирает удаляемые таблицы и нажимает кнопку “сохранить”
  5. Система выводит на экран веб-страницу с обновленным состоянием таблицы с экспериментами
- Альтернативный сценарий:

6. Система выводит на экран веб-страницу со списком уже записанных экспериментов в виде таблицы
7. Пользователь нажимает на кнопку “добавление”/“удаление”
8. Система открывает во всплывающем окне форму для записи эксперимента в таблицу либо список экспериментов с возможностью отметить эксперименты к удалению
9. Пользователь заносит данные либо выбирает удаляемые таблицы и нажимает кнопку “отмена”, так как ввёл неверные данные или выбрал не ту таблицу для удаления
10. Система предлагает пользователю заново ввести данные в открытом окне либо заново выбрать таблицу для удаления
11. Пользователь может вернуться на веб-страницу с таблицей экспериментов, нажав кнопку “отмена”

### **1.2.2. «Экспорт/импорт» результатов проведённого эксперимента в БД**

#### *Основной сценарий:*

1. Система выводит на экран веб-страницу со списком уже записанных экспериментов в виде таблицы
  2. Пользователь нажимает на кнопку “экспорт/импорт”
  3. Система переходит на веб-страницу экспорта/импорта данных эксперимента, на которой можно выбрать формат данных, имя файла с данными (в случае экспорта), выбрать файл с исходными данными среди имеющихся на устройстве пользователя (в случае с импортом)
  4. Пользователь нажимает на кнопку “сохранить”/“открыть”
  5. Система осуществляет экспорт/импорт данных эксперимента с соответствующими параметрами
- Альтернативный сценарий:
6. Система выводит на экран веб-страницу со списком уже записанных экспериментов в виде таблицы
  7. Пользователь нажимает на кнопку “экспорт/импорт”



8. Система переходит на веб-страницу экспорта/импорта данных эксперимента, на которой можно выбрать формат данных, имя файла с данными (в случае экспорта), выбрать файл с исходными данными среди имеющихся на устройстве пользователя (в случае с импортом) Выбранный формат файла не поддерживается
9. Система выводит пользователю соответствующее сообщение об ошибке во всплывающем окне
10. Пользователь может либо повторить попытку экспорта в файл другого формата/импорта другого файла с исходными данными
11. Пользователь может отменить экспорт/импорт, нажав кнопку “отмена”
12. Система возвращает пользователя на веб-страницу с записанными экспериментами

### **1.2.3. Просмотр графических результатов проведённого эксперимента**

*Основной сценарий:*

1. Система выводит на экран веб-страницу со списком уже записанных экспериментов в виде таблицы
2. Пользователь выбирает интересующий его эксперимент путём нажатия на специальную кнопку, расположенную в таблице в поле “просмотр”
3. Система переходит на веб-страницу с графическим представлением результатов эксперимента
4. Пользователь может вернуться на страницу с таблицей всех экспериментов, нажав кнопку “назад”

#### **1.2.4. Поиск эксперимента среди проведённых и занесённых в БД**

*Основной сценарий:*

1. Система выводит на экран веб-страницу со списком уже записанных экспериментов в виде таблицы
2. Пользователь вводит в окне поиска название эксперимента, с результатами которого он хочет ознакомиться
3. Система показывает эксперименты с совпадающими названиями

*Альтернативный сценарий:*

4. Система выводит на экран веб-страницу со списком уже записанных экспериментов в виде таблицы
5. Пользователь вводит в окне поиска название эксперимента, с результатами которого он хочет ознакомиться
6. Система не находит экспериментов с введённым названием и возвращает пользователю сообщение об отсутствии совпадений во всплывающем окне

#### **1.2.5. Просмотр таблицы с данными по проведённому эксперименту**

*Основной сценарий:*

1. Система выводит на экран веб-страницу со списком уже записанных экспериментов в виде таблицы
2. Пользователь выбирает эксперимент из списка путём нажатия на его название
3. Система переходит на веб-страницу с таблицей дронов, которые приняли участие в эксперименте
4. Пользователь имеет возможность добавить/удалить дрон из экспериментов при помощи нажатия соответствующей кнопки. В этом случае система далее будет действовать как в случае с добавлением/удалением данных для эксперимента с той лишь разницей, что начальной страницей будет не список всех экспериментов в виде таблицы, а список дронов в виде таблицы

5. Пользователь имеет возможность экспортировать/импортировать данные по конкретному эксперименту при помощи соответствующих кнопок. В этом случае система далее будет действовать как в случае с экспортом/импортом данных для эксперимента с той лишь разницей, что начальной страницей будет не список всех экспериментов в виде таблицы, а список дронов в виде таблицы

6. Пользователь может ознакомиться с таблицей изменения положения конкретного дрона, нажав на его название. В этом случае система откроет отдельную веб-страницу, где в виде таблицы будут отражены все изменения положения выбранного дрона в ходе эксперимента. Пользователь также имеет возможность удалять и добавлять записи положения выбранного дрона аналогично редактированию всех записей и списка дронов. Пользователь после выполнения желаемых действий нажимает кнопку назад для возврата на страницу со списком всех дронов для данного эксперимента, система переходит на веб-страницу со списком дронов для выбранного ранее эксперимента

7. Пользователь после выполнения желаемых действий нажимает кнопку назад для возврата на страницу со списком всех экспериментов

8. Система переходит на веб-страницу со списком всех экспериментов

## 2. МОДЕЛЬ ДАННЫХ

### 2.1. Описание нереляционной модели

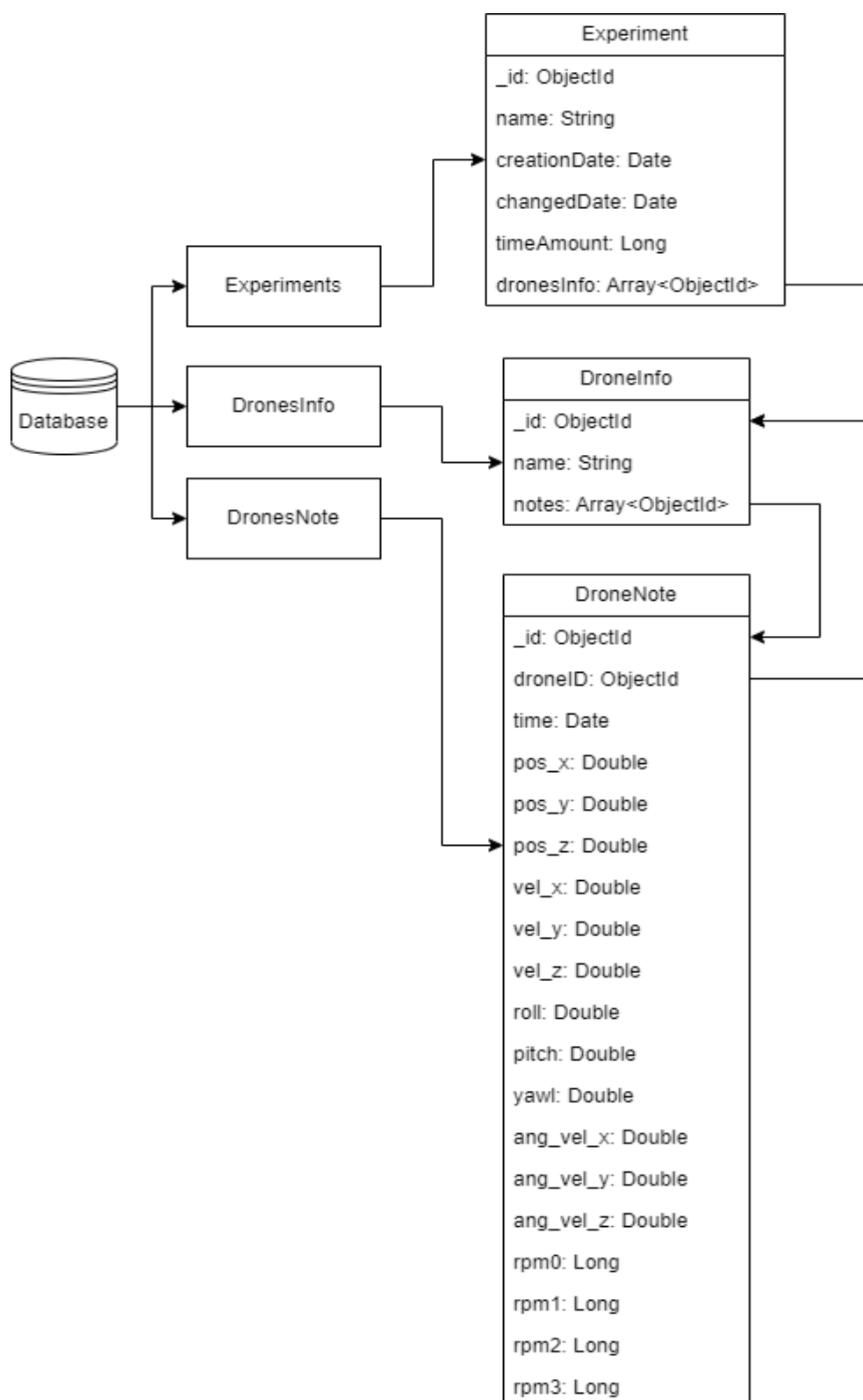


Рис. 2 – Графическое представление модели

БД содержит 3 коллекции - Experiments, DronesInfo, DronesNote – каждая из которых описана ниже.

1. Experiments содержит Experiment

- Experiment - запись об эксперименте
  - `_id` - уникальный идентификатор записи об эксперименте
  - `name` - название эксперимента
  - `creationDate` - дата создания записи об эксперименте
  - `changedDate` - дата изменения записи об эксперименте
  - `timeAmount` - длительность эксперимента
  - `dronesInfo` - информация с дронов(данные дронов)

2. DronesInfo содержит DroneInfo

- DroneInfo - информация с дрона
  - `_id` - уникальный идентификатор информации с дрона
  - `name` - название дрона
  - `notes` - записи измерений с дрона

3. DronesNote содержит DroneNote

- DroneNote - единичное измерение с дрона
  - `_id` - уникальный идентификатор записи измерения с дрона
  - `droneID` - идентификатор дрона, к которому относится запись
  - `time` - дата записи измерения с дрона
  - `pos_x` - позиция дрона по координате x
  - `pos_y` - позиция дрона по координате y
  - `pos_z` - позиция дрона по координате z
  - `vel_x` - скорость дрона по координате x
  - `vel_y` - скорость дрона по координате y
  - `vel_z` - скорость дрона по координате z
  - `roll` - угол крена дрона
  - `pitch` - угол тангажа дрона
  - `yawl` - угол рысканья дрона
  - `rpm0` - частота вращения первой лопасти

- rpm1 - частота вращения второй лопасти
- rpm2 - частота вращения третьей лопасти
- rpm3 - частота вращения четвертой лопасти

### **Оценка объема**

- DroneNote
  - \_id - 12b
  - droneID - 12b
  - time - 8b
  - pos\_x - 8b
  - pos\_y - 8b
  - pos\_z - 8b
  - vel\_x - 8b
  - vel\_y - 8b
  - vel\_z - 8b
  - roll - 8b
  - pitch - 8b
  - yawl - 8b
  - rpm0 - 8b
  - rpm1 - 8b
  - rpm2 - 8b
  - rpm3 - 8b
- DroneInfo
  - \_id - 12b
  - name - до 32b
  - notes - 12b \* notes\_amount
- Experiment
  - \_id - 12b
  - name - до 64b
  - creationDate - 8b

- changedDate - 8b
- timeAmount - 8b
- dronesInfo - 12b \* drones\_amount

Среднее количество записей измерений с дрона - N Среднее количество дронов в эксперименте - 10 Среднее количество экспериментов - 10

### **Фактический объем**

- $V(\text{DroneNote}) = \_id + \text{droneID} + \text{time} + \text{pos\_x} + \text{pos\_y} + \text{pos\_z} + \text{vel\_x} + \text{vel\_y} + \text{vel\_z} + \text{roll} + \text{pitch} + \text{yawl} + \text{rpm0} + \text{rpm1} + \text{rpm2} + \text{rpm3} = 12b * 2 + 8b * 14 = 136b$
- $V\_cp(\text{DroneInfo}) = \_id + \text{name} + 12b * \text{notes\_amount} = 12b + 32b + 12b * \text{notes\_amount} = 44b + 12b * \text{notes\_amount} = 44b + 12b * N$
- $V\_cp(\text{Experiment}) = \_id + \text{name} + \text{creationDate} + \text{changedDate} + \text{timeAmount} + 12b * \text{drones\_amount} = 12b + 64b + 8b * 3 + 12b * \text{drones\_amount} = 100b + 12b * \text{drones\_amount} = 100b + 12b * 10 = 220b$
- $V\_факт = \text{количество\_экспериментов} * (V\_cp(\text{Experiment}) + \text{количество\_дронов} * (V\_cp(\text{DroneInfo}) + \text{количество\_записей} * V(\text{DroneNote}))) = 10 * (220b + 10 * (44b + 12b * N + N * 136b)) = 10 * (660b + 1480N) = 14800N + 6600b$

### **"Чистый" объем**

Чистый объём вычислен на основе фактического, но была исключена избыточность данных:

- $V(\text{DroneNote}) = \text{time} + \text{pos\_x} + \text{pos\_y} + \text{pos\_z} + \text{vel\_x} + \text{vel\_y} + \text{vel\_z} + \text{roll} + \text{pitch} + \text{yawl} + \text{rpm0} + \text{rpm1} + \text{rpm2} + \text{rpm3} = 112b$
- $V\_cp(\text{DroneInfo}) = \_id + 12b * \text{notes\_amount} = 12b + 12b * N$
- $V\_cp(\text{Experiment}) = \_id + 12b * \text{drones\_amount} = 12b + 12b * \text{drones\_amount} = 12b + 12b * 10 = 132b$
- $V\_чист = \text{количество\_экспериментов} * (V\_cp(\text{Experiment}) + \text{количество\_дронов} * (V\_cp(\text{DroneInfo}) + \text{количество\_записей} * V(\text{DroneNote})))$

$$V(\text{DroneNote})) = 10 * (132b + 10 * (12b + 12b * N + N * 112b)) = 12400N + 2520b$$

### Избыточность модели

$$\text{Избыточность} = \text{Фактический объем} / \text{"Чистый" объем} \quad \text{Избыточность} = (14800N + 6600b) / (12400N + 2520b) = 14800N / 12400N = 1.19$$

### Направление роста модели при увеличении количества объектов каждой сущности

При увеличении количества записей измерений с дрона, дронов в эксперименте, числа экспериментов объем данных возрастает линейно. Если одновременно увеличивать  $M$  параметров в  $N$  раз, то рост будет вида  $M * N$

## 2.2. Описание реляционной модели

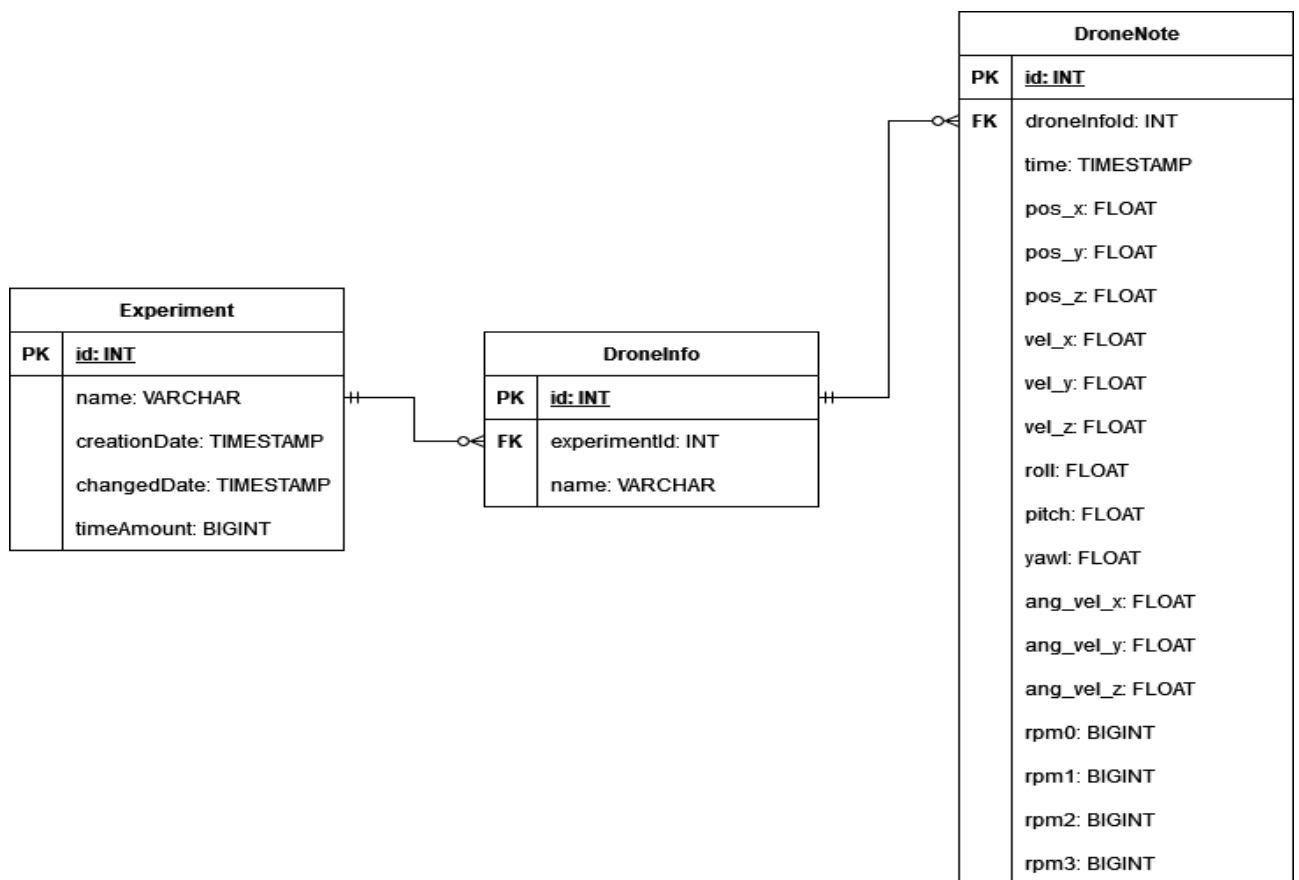


Рис. 3 – Графическое представление модели



Как и нереляционная модель, реляционная модель содержит три коллекции, каждая из которых описана подробно ниже.

- Experiment - запись об эксперименте
  - id - уникальный идентификатор записи об эксперименте
  - name - название эксперимента
  - creationDate - дата создания записи об эксперименте
  - changedDate - дата изменения записи об эксперименте
  - timeAmount - длительность эксперимента
- DroneInfo - информация с дрона
  - id - уникальный идентификатор информации с дрона
  - experimentId - внешний ключ для связи с экспериментом
  - name - название дрона
- DroneNote - единичное измерение с дрона
  - id - уникальный идентификатор записи измерения с дрона
  - droneInfoId - внешний ключ для связи с информацией с дрона
  - time - дата записи измерения с дрона
  - pos\_x - позиция дрона по координате x
  - pos\_y - позиция дрона по координате y
  - pos\_z - позиция дрона по координате z
  - vel\_x: скорость дрона по координате x
  - vel\_y: скорость дрона по координате y
  - vel\_z: скорость дрона по координате z
  - roll - угол крена дрона
  - pitch - угол тангажа дрона
  - yawl - угол рысканья дрона
  - rpm0 - частота вращения первой лопасти
  - rpm1 - частота вращения второй лопасти
  - rpm2 - частота вращения третьей лопасти
  - rpm3 - частота вращения четвертой лопасти

## Оценка объема

- DroneNote
  - id - 4b
  - droneInfoId - 4b
  - time - 8b
  - pos\_x - 8b
  - pos\_y - 8b
  - pos\_z - 8b
  - vel\_x: 8b
  - vel\_y: 8b
  - vel\_z: 8b
  - roll - 8b
  - pitch - 8b
  - yawl - 8b
  - rpm0 - 8b
  - rpm1 - 8b
  - rpm2 - 8b
  - rpm3 - 8b
- DroneInfo
  - id - 4b
  - experimentId - 4b
  - name - до 32b
- Experiment
  - id - 4b
  - name - до 64b
  - creationDate - 8b
  - changedDate - 8b
  - timeAmount - 8b

Среднее количество записей измерений с дрона - N Среднее количество дронов в эксперименте - 10 Среднее количество экспериментов – 10

### **Фактический объем**

- $V(\text{DroneNote}) = \text{id} + \text{droneInfoId} + \text{time} + \text{pos\_x} + \text{pos\_y} + \text{pos\_z} + \text{vel\_x} + \text{vel\_y} + \text{vel\_z} + \text{roll} + \text{pitch} + \text{yaw} + \text{rpm0} + \text{rpm1} + \text{rpm2} + \text{rpm3} = 4b * 2 + 8b * 14 = 120b$
- $V(\text{DroneInfo}) = \text{id} + \text{experimentId} + \text{name} = 4b * 2 + 32b = 40b$
- $V(\text{Experiment}) = \text{id} + \text{name} + \text{creationDate} + \text{changedDate} + \text{timeAmount} = 4b + 64b + 8b * 3 = 92b$
- $V_{\text{факт}} = \text{количество\_экспериментов} * (V(\text{Experiment}) + \text{количество\_дронов} * (V(\text{DroneInfo}) + \text{количество\_записей} * V(\text{DroneNote}))) = 10 * (92b + 10 * (40b + N * 120b)) = 10 * (492b + 1200N) = 12000N + 4920b$

### **"Чистый" объем**

- $V(\text{DroneNote}) = \text{time} + \text{droneInfoId} + \text{pos\_x} + \text{pos\_y} + \text{pos\_z} + \text{vel\_x} + \text{vel\_y} + \text{vel\_z} + \text{roll} + \text{pitch} + \text{yaw} + \text{rpm0} + \text{rpm1} + \text{rpm2} + \text{rpm3} = 4b + 8b * 14 = 116b$
- $V_{\text{cp}}(\text{DroneInfo}) = \text{id} + \text{experimentId} = 8b$
- $V_{\text{cp}}(\text{Experiment}) = \text{id} = 4b$
- $V_{\text{чист}} = \text{количество\_экспериментов} * (V(\text{Experiment}) + \text{количество\_дронов} * (V(\text{DroneInfo}) + \text{количество\_записей} * V(\text{DroneNote}))) = 10 * (4b + 10 * (8b + N * 116b)) = 11600N + 840b$

### **Избыточность модели**

Избыточность = Фактический объем / "Чистый" объем  
 $\text{Избыточность} = (12000N + 4920b) / (11600N + 840b) = 12000N / 11600N = 1.03$

## **Направление роста модели при увеличении количества объектов каждой сущности**

При увеличении количества записей измерений с дрона, дронов в эксперименте, числа экспериментов объем данных возрастает линейно. Если одновременно увеличивать  $M$  параметров на  $N$  раз, то рост будет вида  $M * N$ .

### **2.3. Сравнение моделей**

#### **Удельный объем информации**

$$V_{\text{факт\_нерел}} = 14800N + 6600b \quad V_{\text{факт\_рел}} = 12000N + 4920b$$

$V_{\text{факт\_рел}} / V_{\text{факт\_нерел}} = 0.81$ . Видно, что объем информации при прочих равных условиях в реляционной модели меньше на 19%. Также избыточность модели нереляционной модели больше -  $1.19 > 1.03$ .

#### **Запросы по отдельным юзкейсам**

Нереляционная модель:

- Добавление результатов проведенного эксперимента в БД -  $3 + \text{update} * N_{\text{дронов}}$
- Удаление результатов проведенного эксперимента в БД - 5
- Поиск эксперимента среди проведенных и занесенных в БД - 1
- Просмотр таблицы с данными по проведенному эксперименту - 3

Реляционная модель:

- Добавление результатов проведенного эксперимента в БД - 5
- Удаление результатов проведенного эксперимента в БД - 5
- Поиск эксперимента среди проведенных и занесенных в БД - 1
- Просмотр таблицы с данными по проведенному эксперименту - 3

### **Количество задействованных коллекций**

Нереляционная модель - 3:

- Experiments
- DronesInfo
- DronesNote

Реляционная модель - 3:

- Experiment
- DroneInfo
- DroneNote

### **Вывод по итогам сравнения**

Реляционная модель превосходит нереляционную по объему информации. По запросам они одинаковы, кроме запроса на добавление, т.к. требуется дополнительно узнать id добавленных строк в реляционной модели.

### 3. РАЗРАБОТАННОЕ ПРИЛОЖЕНИЕ

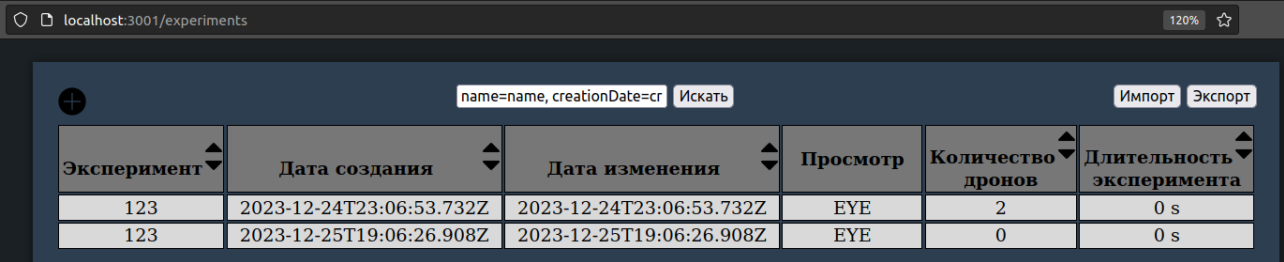
#### 3.1. Краткое описание

Разработанное приложение имеет клиент-серверную архитектуру и использует REST-подход.

Приложение написано на языке программирования JavaScript. Клиентская часть реализована при помощи HTML и CSS, серверная часть – при помощи фреймворка express, данные хранятся в коллекциях в MongoDB.

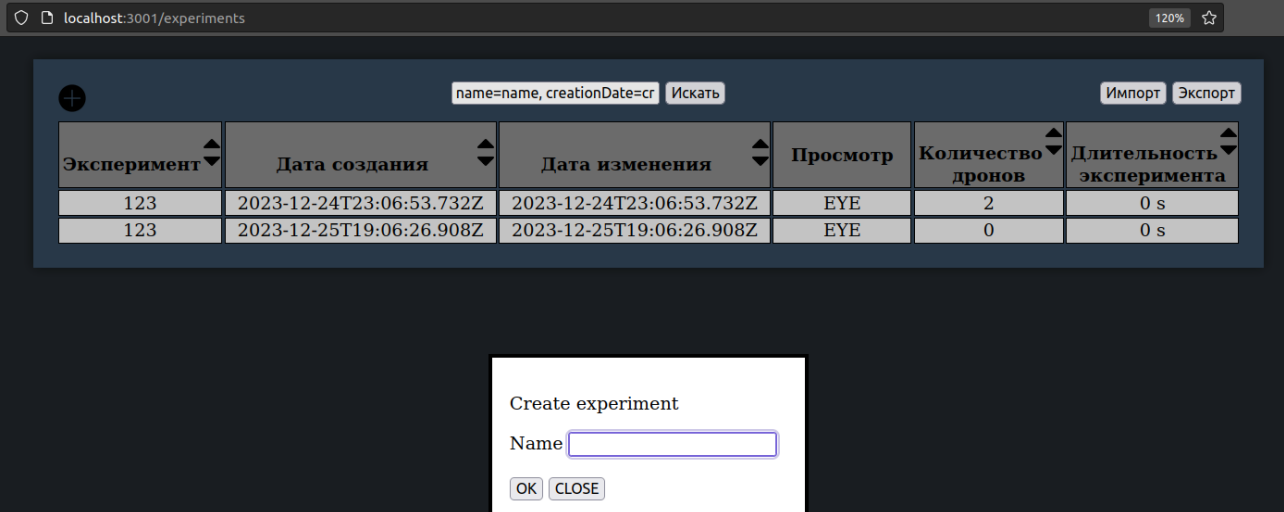
Приложение разворачивается при помощи платформы докер, используя три отдельных виртуальных контейнера для клиентской части приложения, серверной части приложения и базы данных.

#### 3.2. Снимки экрана приложения



Эксперимент	Дата создания	Дата изменения	Просмотр	Количество дронов	Длительность эксперимента
123	2023-12-24T23:06:53.732Z	2023-12-24T23:06:53.732Z	EYE	2	0 s
123	2023-12-25T19:06:26.908Z	2023-12-25T19:06:26.908Z	EYE	0	0 s

Рис. 4 – Страница со списком всех экспериментов



Create experiment

Name

OK CLOSE

Рис. 5 – Страница с добавлением нового эксперимента

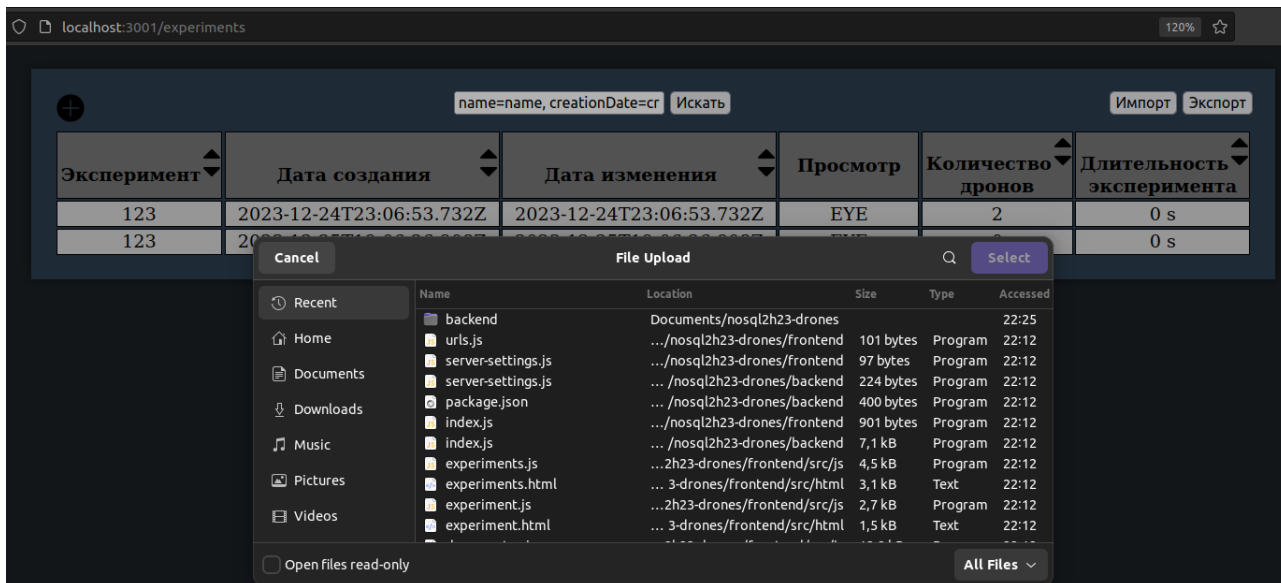


Рис. 6 – Страница «Импорт данных»

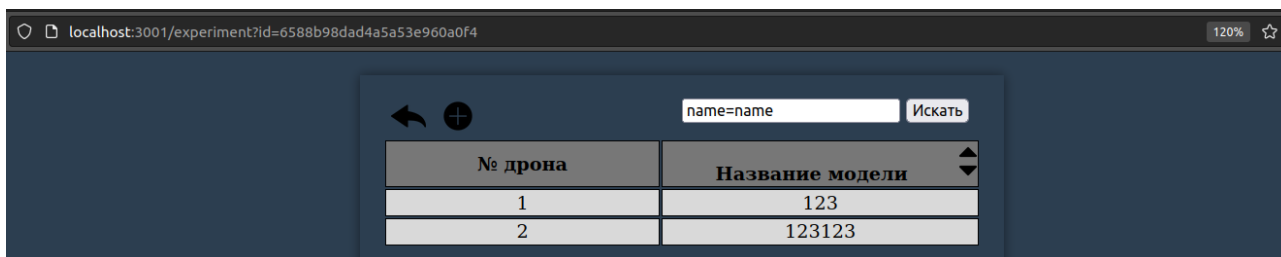


Рис. 7 – Страница со списком всех дронов для эксперимента

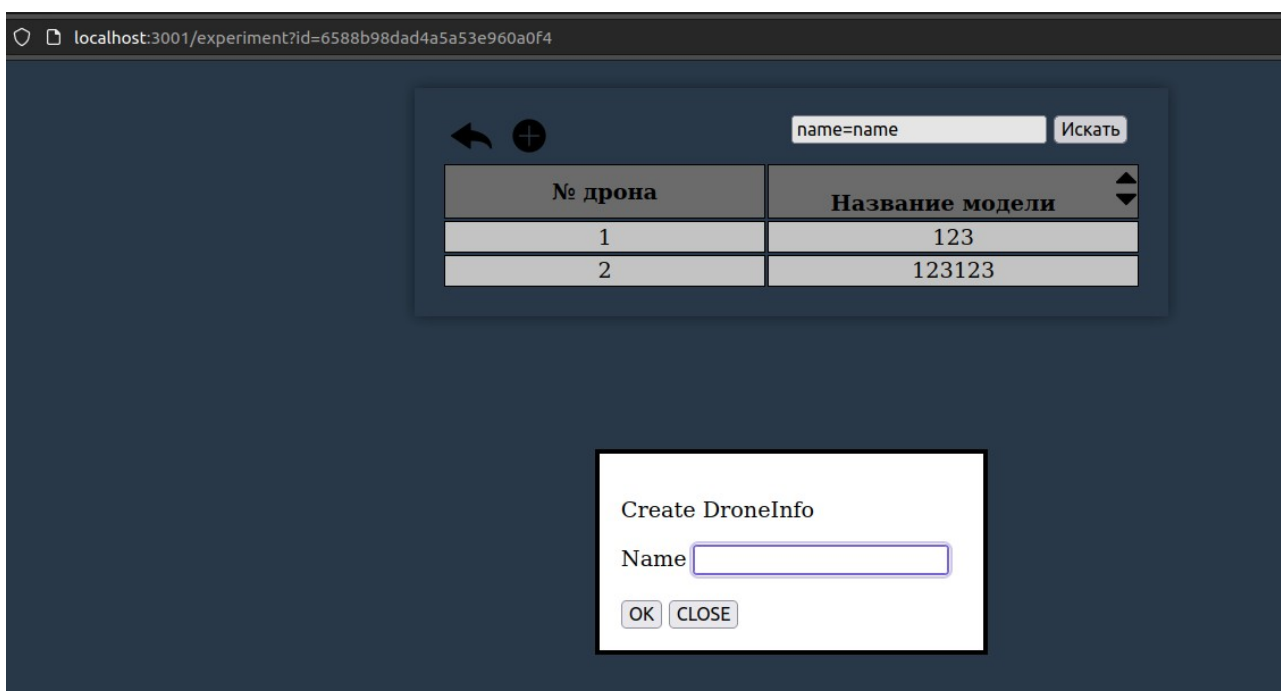


Рис. 8 – Страница с добавлением нового дрона в эксперимент

Time	pos_x	pos_y	pos_z	vel_x	vel_y	vel_z	roll	pitch	yaw	ang_vel_x	ang_vel_y	ang_vel_z	rpm0	rpm1	rpm2	rpm3
2001-01-01T00:00:00.000Z	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2001-02-01T00:00:00.000Z	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2

Рис. 9 – Страница с данными для дрона в выбранном эксперименте

localhost:3001/drone-notes?experiment\_id=6588b98dad4a5a53e960a0f4&drone\_info\_id=6588c19b869f0578aabbf7d1

time=time, pos\_x=pos\_x, pos\_y=pos\_y Искать

Time	pos_x	pos_y	pos_z	vel_x
2001-01-01T00:00:00.000Z	1	1	1	1
2001-02-01T00:00:00.000Z	2	2	2	2

Create DroneNote

Time

pos\_x

pos\_y

pos\_z

vel\_x

vel\_y

vel\_z

roll

pitch

yaw

ang\_vel\_x

ang\_vel\_y

ang\_vel\_z

rpm0

rpm1

rpm2

rpm3

OK CLOSE

time=time, pos\_x=pos\_x, pos\_y=pos\_y Искать

ang_vel_x	ang_vel_y	ang_vel_z	rpm0	rpm1	rpm2	rpm3
1	1	1	1	1	1	1
2	2	2	2	2	2	2

Активация Windows  
Чтобы активировать Windows, перейдите на [www.microsoft.com/windows/activation](#)

Рис. 10 – Страница с добавлением данных дрона в эксперимент



## ЗАКЛЮЧЕНИЕ

В ходе работы над ИДЗ было разработано веб-приложение, осуществляющее хранение данных об экспериментах, проведённых в эмуляторе pybullet\_drones. Была реализована следующая функциональность: импорт/экспорт данных, сортировка и фильтрация данных, визуальное представление и просмотр статистики данных. Также были освоены и применены основные принципы работы с docker`ом и MongoDB.

В качестве недостатков полученного решения можно отметить примитивный дизайн, а также недостаточно широкий функционал (отсутствие возможности посмотреть графики движения дронов в эмуляторе).

В перспективе развития прежде всего следует доработать внешний вид приложения, а также расширить функционал: реализовать хранение данных с возможностью удаленного доступа, реализовать графическое представление движения дрона в эксперименте.