

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение высшего образования
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ АЭРОКОСМИЧЕСКОГО
ПРИБОРОСТРОЕНИЯ»

КАФЕДРА КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ И ПРОГРАММНОЙ ИНЖЕНЕРИИ

КУРСОВОЙ ПРОЕКТ
ЗАЩИЩЕН С ОЦЕНКОЙ
РУКОВОДИТЕЛЬ

Старший преподаватель

должность, уч. степень, звание

подпись, дата

Т. И. Белая

инициалы, фамилия

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА К КУРСОВОМУ ПРОЕКТУ

Разработка информационной системы управления беспилотными
устройствами для охраны территорий

по курсу: Проектирование программных систем

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. №

4133

подпись, дата

Ковалев Д.В

инициалы, фамилия

Санкт-Петербург 2024

РЕФЕРАТ

Отчет 56 с., 15 рис., 2 прил.

АВТОМАТИЗИРОВАННАЯ СИСТЕМА УПРАВЛЕНИЯ И АНАЛИЗА ДАННЫХ, ОХРАНЫ ТЕРРИТОРИЙ, БАЗА ДАННЫХ, ФУНКЦИИ, МОДЕЛИРОВАНИЕ

Объектом исследования является процесс управления охранными мероприятиями, включающий функции управления, мониторинга и анализа данных с различных датчиков, установленных на БПЛА.

Цель работы – разработка автоматизированной информационной системы, которая будет оптимизировать деятельность охранных мероприятий на территориях.

В процессе работы было проведено исследование предметной области, которое позволило выяснить требования и границы информационной системы.

В результате работы была создана автоматизированная система для управления и анализа данных с БПЛА, обеспечивающая повышение эффективности и продуктивности охранных мероприятий.

СОДЕРЖАНИЕ

РЕФЕРАТ.....	2
СОДЕРЖАНИЕ	3
ВВЕДЕНИЕ.....	5
1. ЦЕЛИ И НАЗНАЧЕНИЕ СОЗДАНИЯ АВТОМАТИЗИРОВАННОЙ	6
1.1 Назначение программы	6
1.2 Бизнес-функции, для автоматизации которых предназначена система....	6
2. ПРОЕКТИРОВАНИЕ И КОМПЛЕКСНОЕ МОДЕЛИРОВАНИЕ	7
2.1 Характеристика и структура программы.....	7
2.2 Описание основных особенностей программы	7
2.3 Основные пользователи системы	8
2.4 Функциональное моделирование в методике IDEF0.....	9
2.4.1 Контекстная диаграмма	9
2.4.2 Декомпозиция контекстной диаграммы	10
2.4.3 Декомпозированная диаграмма второго уровня ветки A1	11
2.4.4 Декомпозированная диаграмма второго уровня ветки A2	12
2.4.5 Декомпозированная диаграмма второго уровня ветки A3	13
2.4.6 Декомпозированная диаграмма второго уровня ветки A4	14
2.5 Моделирование в методике DFD	15
2.6 Объектное моделирование в методике UML.....	16
2.6.1 Диаграмма вариантов использования	16
2.6.2 Диаграмма компонентов.....	17
2.6.3 Диаграмма пакетов.....	18
2.6.4 Диаграмма размещения	19
2.7 Проектирование базы данных.....	20
2.7.1 Словарь данных.....	21
3. ТРЕБОВАНИЯ К ИНФОРМАЦИОННОЙ СИСТЕМЕ	24
3.1 Требования к структуре АС в целом.....	24
3.2 Функциональные требования к системе.....	24

3.2.1	Просмотр, редактирование и удаление Секторов и Дронов.....	24
3.2.2	Добавление событий с дронов	26
3.2.3	Анализ данных и формирования отчета с данными с дронов	27
4.	РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ	28
4.1	Базы данных.....	28
4.2	Приложение клиент	28
	ЗАКЛЮЧЕНИЕ	30
	СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	32
	ПРИЛОЖЕНИЕ А.....	33
	ПРИЛОЖЕНИЕ Б	43

ВВЕДЕНИЕ

В современных условиях цифровизации различных отраслей экономики, необходимость в эффективных системах управления и анализа данных становится все более очевидной. Это касается и сферы охраны, где внедрение информационных систем позволяет значительно повысить продуктивность и эффективность охранных мероприятий. Курсовой проект посвящен разработке информационной системы управления и анализа данных с БПЛА, что является актуальным направлением в сфере охраны территорий.

Цель данного проекта – разработка автоматизированной информационной системы, которая позволит охранным агентствам эффективно собирать, хранить и анализировать данные, полученные с различных датчиков, установленных на БПЛА. Такая система обеспечит возможность ретроспективного мониторинга территорий, что позволит оперативно реагировать на изменения условий и принимать обоснованные решения по управлению охранными мероприятиями.

1. ЦЕЛИ И НАЗНАЧЕНИЕ СОЗДАНИЯ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ

1.1 Назначение программы

Разрабатываемая информационная система предназначена для автоматизации процессов управления, мониторинга и анализа данных с БПЛА. Она позволяет получать данные с различных оптических источников, таких как камеры, датчики и хранить эти данные в централизованной базе данных. Анализ собранных данных позволяет принимать обоснованные решения по управлению охранными мероприятиями, такими как планирование, патрулирование и обнаружение опасностей.

1.2 Бизнес-функции, для автоматизации которых предназначена система

Функциональность системы позволяет выполнять следующие операции:

- **Хранение исторических данных:** централизованное хранение данных о БПЛА и патрулях для последующего анализа и отчетности.
- **Анализ данных:** предоставление инструментов для анализа данных, таких как сводные таблицы, что позволяет выявлять тенденции и аномалии в изменениях на территориях.
- **Управление БПЛА и секторами:** добавление, редактирование и удаление информации о секторах и БПЛА на территориях через удобный интерфейс администратора.

2. ПРОЕКТИРОВАНИЕ И КОМПЛЕКСНОЕ МОДЕЛИРОВАНИЕ СИСТЕМЫ

2.1 Характеристика и структура программы

Информационная система для управления и анализа данных с БПЛА была разработана с целью повышения эффективности управления охранных мероприятий за счет автоматизации управления, мониторинга и анализа данных, с применением различных БПЛА, развёрнутых на территориях.

Структура программы включает следующие основные компоненты:

- **Декстопная часть:** реализована на языке программирования C++ с использованием фреймворка QT. В десктопной части реализованы основные функции, отвечающие за обработку данных, взаимодействие с базой данных и предоставление функционала для интерфейса приложения.
- **База данных:** используется PostgreSQL 15, которая обеспечивает надежное хранение и быстрый доступ к данным. База данных структурирована таким образом, чтобы эффективно сохранять и обрабатывать данные системы, а также поддерживать систему авторизацию пользователей.
- **GUI:** интерфейс, реализованный с использованием фреймворка QT. GUI обеспечивает удобный доступ к функционалу системы для пользователей.

2.2 Описание основных особенностей программы

Основные особенности программы включают:

- **Монолитная архитектура:** все компоненты системы объединены в единое приложение, что упрощает разработку и развертывание системы.
- **Централизованное хранение данных:** все данные системы хранятся в единой базе данных PostgreSQL, что обеспечивает быстрый доступ и надежность хранения.
- **Инструменты анализа данных:** система включает функционал для построения сводных таблиц, что позволяет пользователям проводить анализ

собранных данных, выявлять тенденции и аномалии в изменениях на территориях.

- Управление БПЛА и секторами: добавление, редактирование и удаление информации о секторах и БПЛА на территориях, что обеспечивает гибкость и адаптивность системы под конкретные нужды предприятия.

2.3 Основные пользователи системы

Основными пользователями системы являются:

- Администраторы: используют систему для мониторинга состояния на территориях, анализа данных системы и принятия решений по управлению охранными мероприятиями. Администраторы имеют доступ к данным и управленческим инструментам системы.

- Операторы: занимаются управлением и мониторингом работы БПЛА во время патрулей. Операторы используют систему для управления состоянием БПЛА, настройкой оборудования, а также сбором информации.

- Аналитики: используют систему для получения сводных отчетов о состоянии на территории и эффективности охранных мероприятий. Аналитики могут формировать стратегические решения на основе данных, предоставляемых системой.

Система разработана таким образом, чтобы каждый пользователь имел доступ только к необходимым ему функциям, что обеспечивает безопасность и удобство использования.

2.4 Функциональное моделирование в методике IDEF0

IDEF0 используется для создания функциональной модели, отображающей структуру и функции системы, а также потоки информации и материальных объектов, связывающие эти функции

2.4.1 Контекстная диаграмма

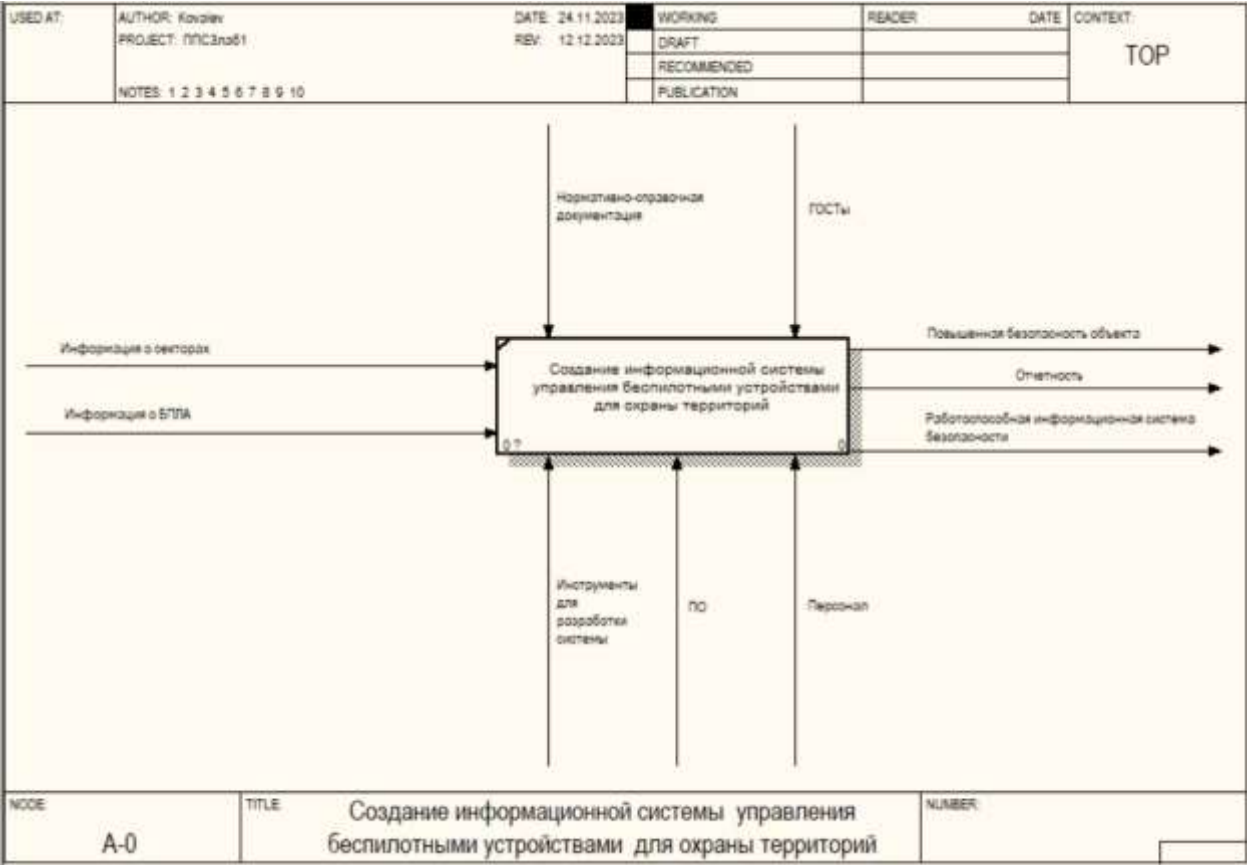


Рисунок 1 - Контекстная диаграмма

2.4.2 Декомпозиция контекстной диаграммы

Декомпозиция контекстной диаграммы в методологии IDEF0 — это процесс детализации и разделения высокого уровня представления системы на более мелкие и управляемые части. Контекстная диаграмма (или A0 диаграмма) в IDEF0 показывает основную функцию системы, взаимодействие системы с внешними объектами (входы, выходы, механизмы и управления), но не вдаётся в детали внутренней структуры системы. В данном случае мы декомпозировали систему на 4 основных задачи.

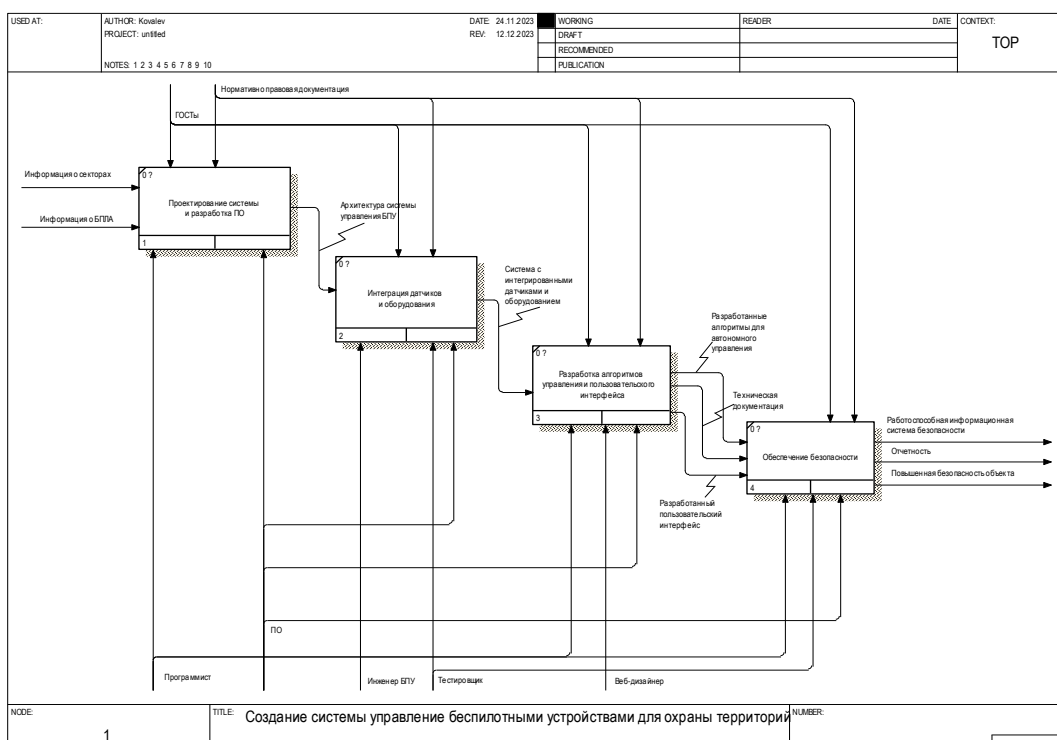


Рисунок 2 – Декомпозированная контекстная диаграмма

2.4.3 Декомпозированная диаграмма второго уровня ветки A1

Декомпозиция задачи «Проектирование системы и разработка ПО» на 3 подзадачи.

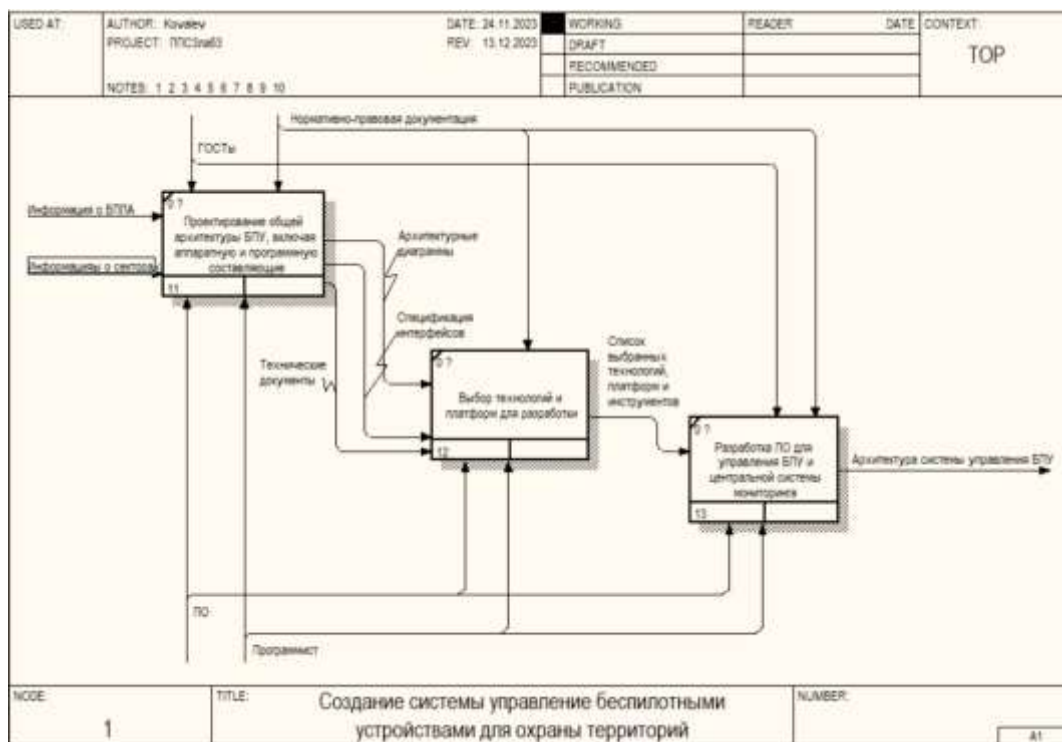


Рисунок 3 - Декомпозированная диаграмма второго уровня ветки A1

2.4.4 Декомпозированная диаграмма второго уровня ветки A2

Декомпозиция задачи «интеграция датчиков и оборудования» на 3 подзадачи.

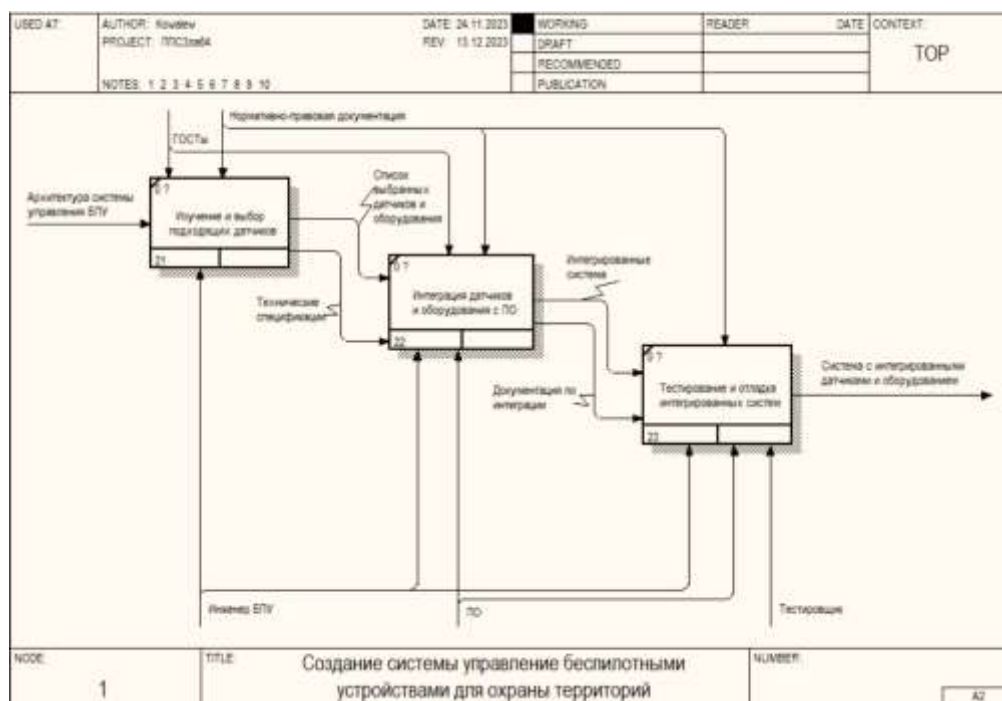


Рисунок 4 - Декомпозированная диаграмма второго уровня ветки A1

2.4.5 Декомпозированная диаграмма второго уровня ветки А3

Декомпозиция задачи «разработка алгоритмов управления пользовательского интерфейса» на 3 подзадачи.

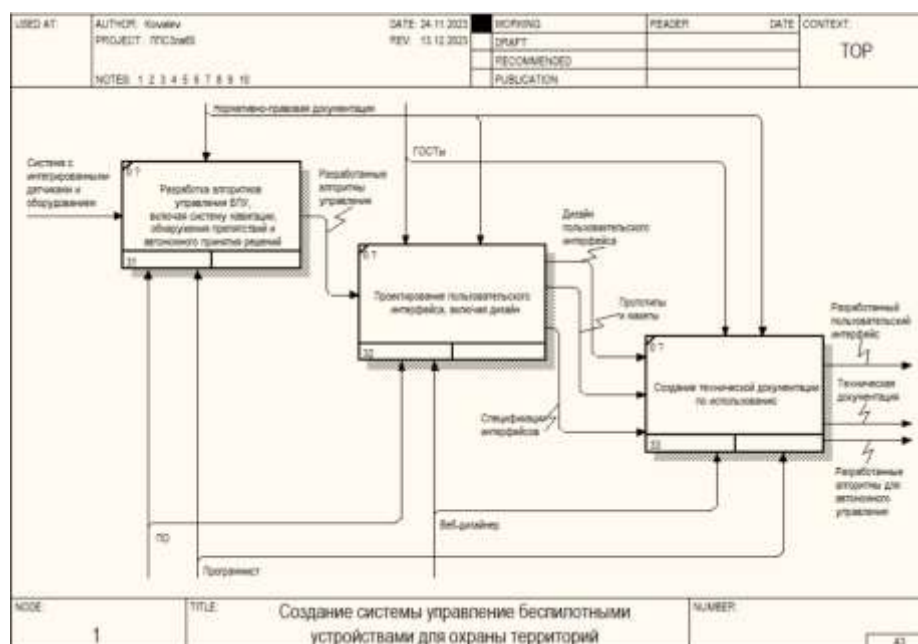


Рис. 5 – Декомпозированная диаграмма второго уровня ветки А3

2.4.6 Декомпозированная диаграмма второго уровня ветки А4

Декомпозиция задачи «обеспечение безопасности» на 3 подзадачи.

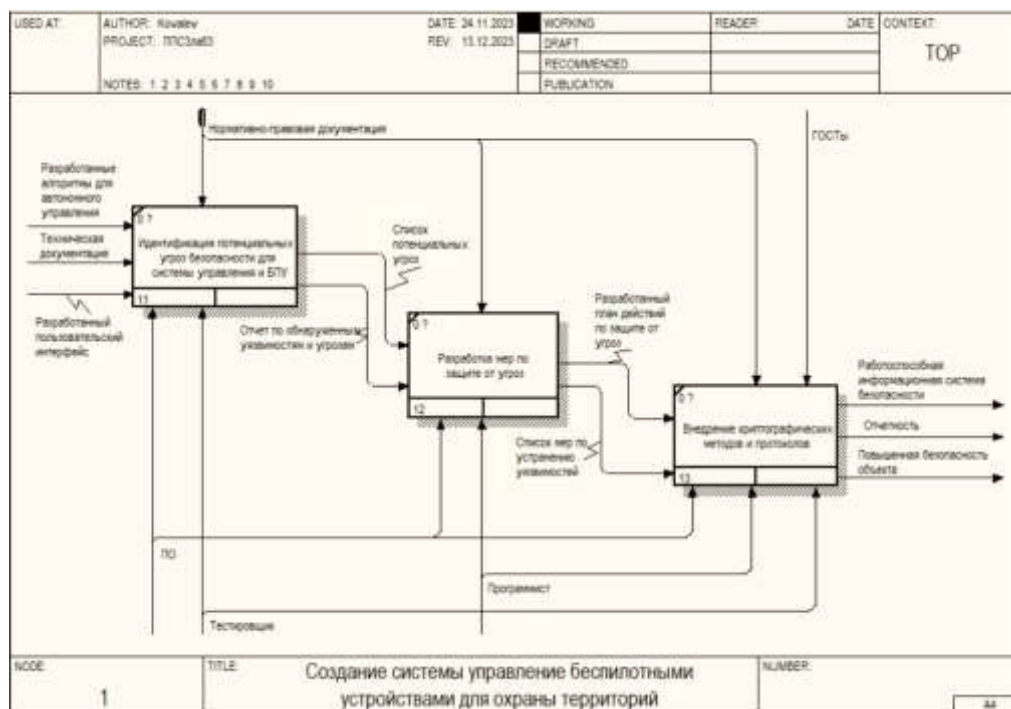


Рис. 6 – Декомпозированная диаграмма второго уровня ветки А4

2.5 Моделирование в методике DFD

DFD диаграммы в отличие от других нотаций позволяют визуально показать все процессы с точки зрения данных. DFD-модели могут быть использованы в дополнение к модели IDEF0 для более наглядного отображения текущих операций документооборота в корпоративных системах обработки информации. Диаграммы потоков данных являются основным средством моделирования функциональных требований к проектируемой системе.

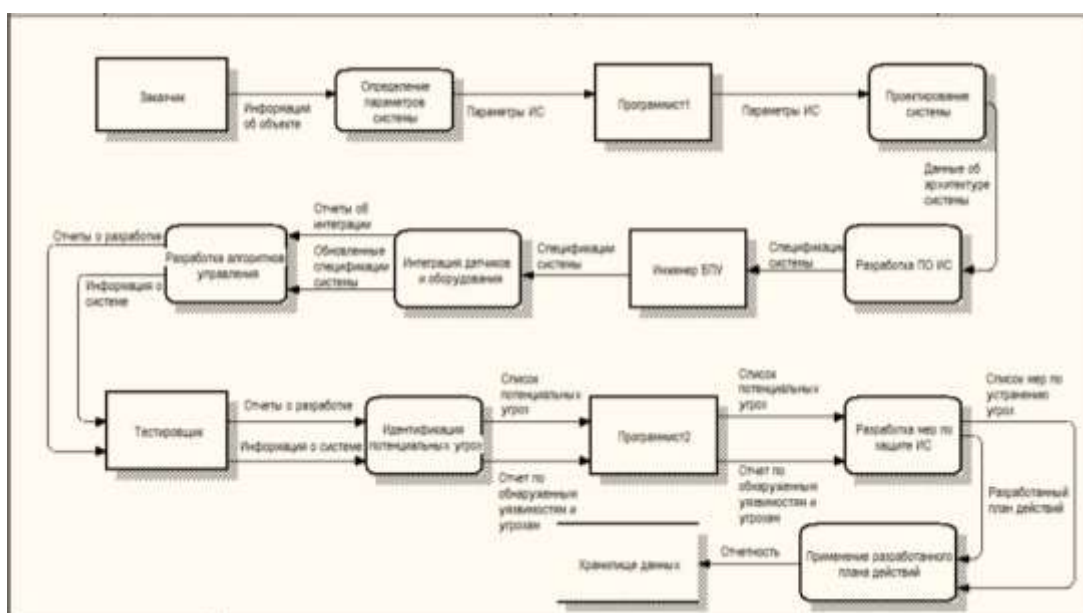


Рисунок 7 – Диаграмма потоковых данных

Рисунок 8 - Диаграмма вариантов использования

2.6.2 Диаграмма компонентов

Диаграмма компонентов UML используется для визуализации и понимания структуры программной системы на уровне высокоуровневых компонентов и их взаимосвязей. Она отображает различные программные компоненты, такие как модули, библиотеки, и их интерфейсы, а также зависимости между ними. Эта диаграмма помогает определить, как различные части системы взаимодействуют друг с другом, выявить потенциальные проблемы с интеграцией и зависимостями, и спланировать модульность и повторное использование компонентов. Благодаря диаграммам компонентов можно эффективно управлять сложностью системы, улучшать её модульность и способствовать ясному разделению ответственности между различными частями проекта.

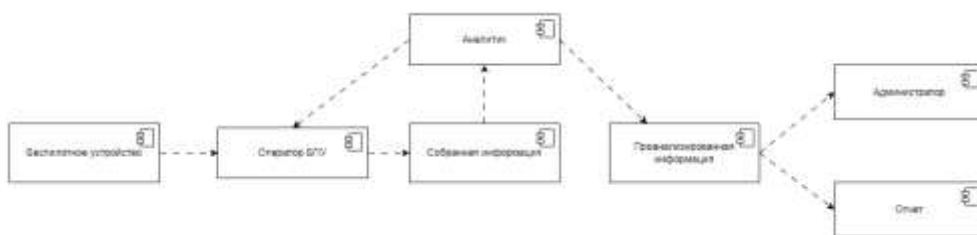


Рисунок 9 - Диаграмма компонентов

2.6.3 Диаграмма пакетов

Диаграмма пакетов UML используется для организации и группировки различных элементов модели, в логически связанные группы, облегчая понимание и управление сложными системами. Эта диаграмма помогает визуализировать структуру системы на более высоком уровне абстракции, показывая, как пакеты взаимодействуют друг с другом через зависимости и отношения.

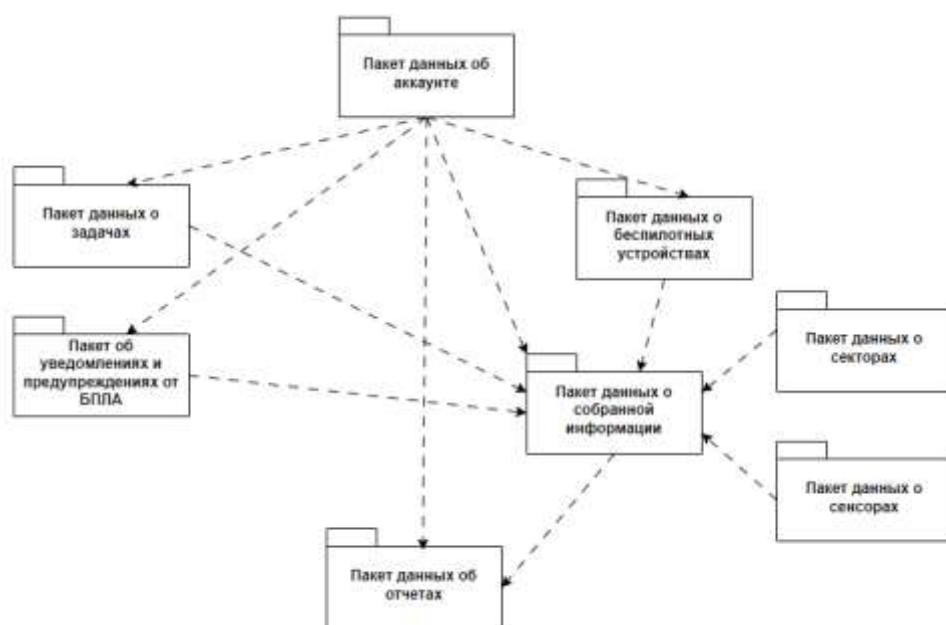


Рисунок 10 - Диаграмма пакетов

2.6.4 Диаграмма размещения

Диаграмма размещения UML используется для моделирования физического развёртывания артефактов программного обеспечения на узлах аппаратного обеспечения. Она показывает, как программные компоненты и сервисы распределены по физическим устройствам, таким как серверы, компьютеры и другие аппаратные узлы, а также описывает связи и взаимодействия между ними. Эта диаграмма полезна для визуализации и анализа архитектуры системы с точки зрения её физической реализации, что помогает в планировании развёртывания, оценке производительности, надёжности и масштабируемости системы.

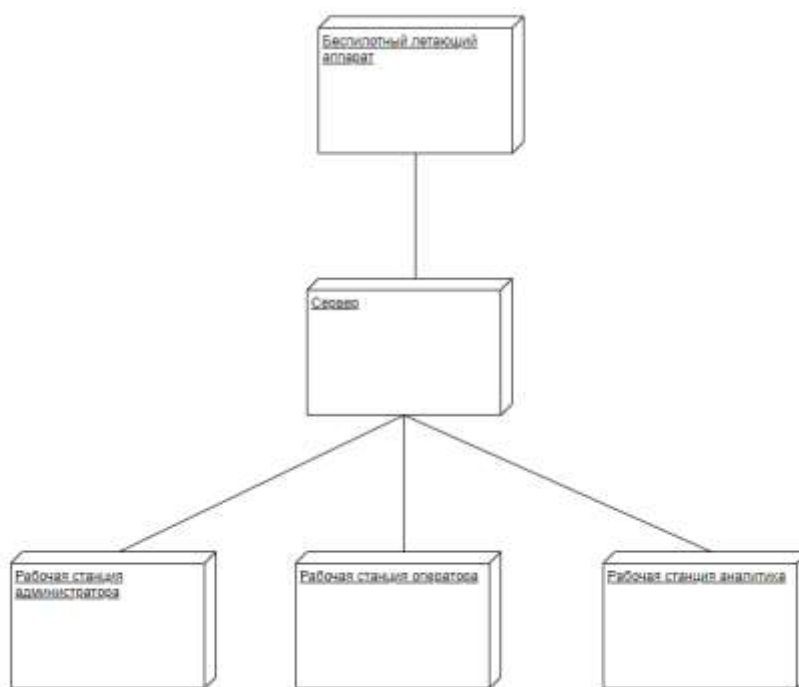


Рисунок 11 - Диаграмма размещения

2.7 Проектирование базы данных

2.7.1 Логическая модель данных

Логическая модель базы данных используется для представления структуры данных системы на концептуальном уровне, независимо от конкретной СУБД. Логическая модель помогает четко определить требования к данным и их структуру, обеспечить целостность и нормализацию данных, а также устранить избыточность.

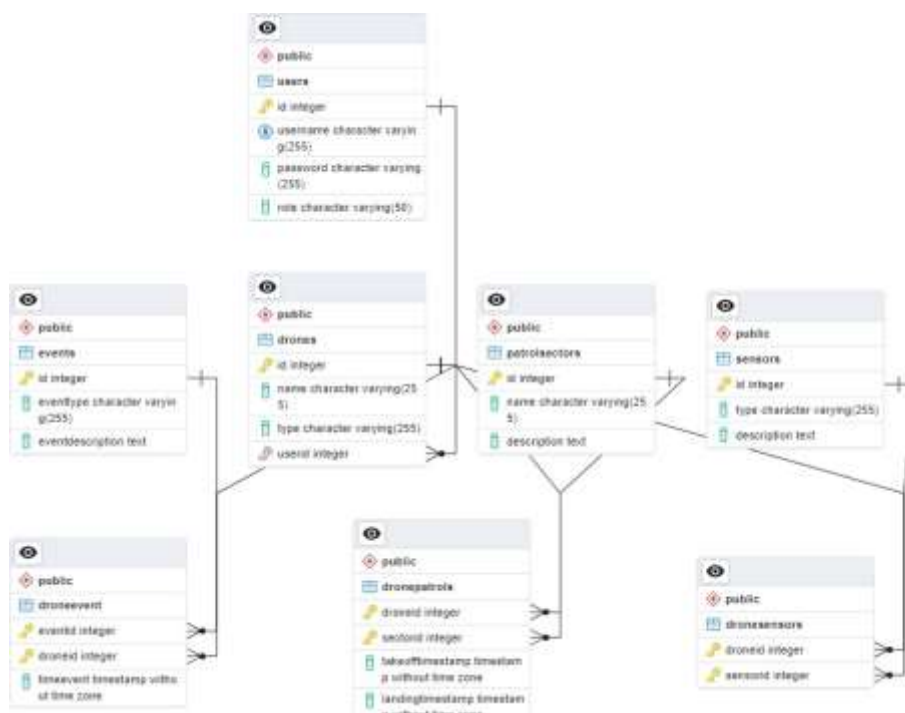


Рисунок 12 - Модель базы данных

2.7.1 Словарь данных

Наименование	Описание
id	Целочисленный тип до 4 байт. Уникальный идентификатор пользователя.
Username character varying	Строковый тип, длина до 255 символов. Имя пользователя
Password character varying	Строковый тип, длина до 255 символов. Пароль пользователя
Role character varying	Строковый тип, длина до 50 символов. Роль пользователя
id	Целочисленный тип до 4 байт. Уникальный идентификатор дрона.
name character varying	Строковый тип, длина до 255 символов. Имя дрона
Type character varying	Строковый тип, длина до 255 символов. Тип дрона
Userid	Целочисленный тип до 4 байт. Уникальный идентификатор пользователя привязанного к дрону.
id	Целочисленный тип до 4 байт. Уникальный идентификатор сектора
name character varying	Строковый тип, длина до 255 символов. Имя сектора
description	Строковый тип, длина до 65535 символов. Описание сектора
id	Целочисленный тип до 4 байт. Уникальный идентификатор сенсора

name character varying	Строковый тип, длина до 255 символов. Имя сенсора
description	Строковый тип, длина до 65535 символов. Описание сенсора
id	Целочисленный тип до 4 байт. Уникальный идентификатор события
name character varying	Строковый тип, длина до 255 символов. Имя события
eventdescription	Строковый тип, длина до 65535 символов. Описание события
eventid	Целочисленный тип до 4 байт. Уникальный идентификатор события
droneid	Целочисленный тип до 4 байт. Уникальный идентификатор дрона.
Timeevent timestamp	Тип дата, формат день, месяц, год, арабскими цифрами, разделенными точкой — 01.01.2020.
droneid	Целочисленный тип до 4 байт. Уникальный идентификатор дрона.
sectorid	Целочисленный тип до 4 байт. Уникальный идентификатор сектора
takeofftimestamp	Тип дата, формат день, месяц, год, арабскими цифрами, разделенными точкой — 01.01.2020.
landingtimestamp	Тип дата, формат день, месяц, год, арабскими цифрами, разделенными точкой — 01.01.2020.

droneid	
sensorid	Целочисленный тип до 4 байт. Уникальный идентификатор сенсора

3. ТРЕБОВАНИЯ К ИНФОРМАЦИОННОЙ СИСТЕМЕ

3.1 Требования к структуре АС в целом

Информационная система для автоматизации процессов управления, мониторинга и анализа данных с БПЛА должна обеспечивать эффективное и надежное выполнение всех необходимых функций. В данном разделе описаны основные требования к структуре системы, необходимой для ее корректного функционирования.

3.2 Функциональные требования к системе

3.2.1 Просмотр, редактирование и удаление Секторов и Дронов

Пользователь должен иметь возможность просматривать информацию о существующих секторах и дронах, добавлять новые сектора и дроны, редактировать данные существующих секторов и дронов и удалять ненужные записи. На странице должна быть представлена таблица с перечнем всех секторов и дронов. Для добавления нового сектора пользователь вводит название сектора и его описание, и нажимает кнопку "Добавить ". Для добавления нового дрона пользователь вводит название дрона, его тип и привязанного оператора, и нажимает кнопку "Добавить ". Для редактирования существующего поля пользователь выбирает нужное поле в таблице и нажимает кнопку "Редактировать", после чего открывается новая форма для ввода. Для удаления поля пользователь выбирает его в таблице и нажимает кнопку "Удалить", после чего система запрашивает подтверждение действия и, при согласии, удаляет запись из базы данных.

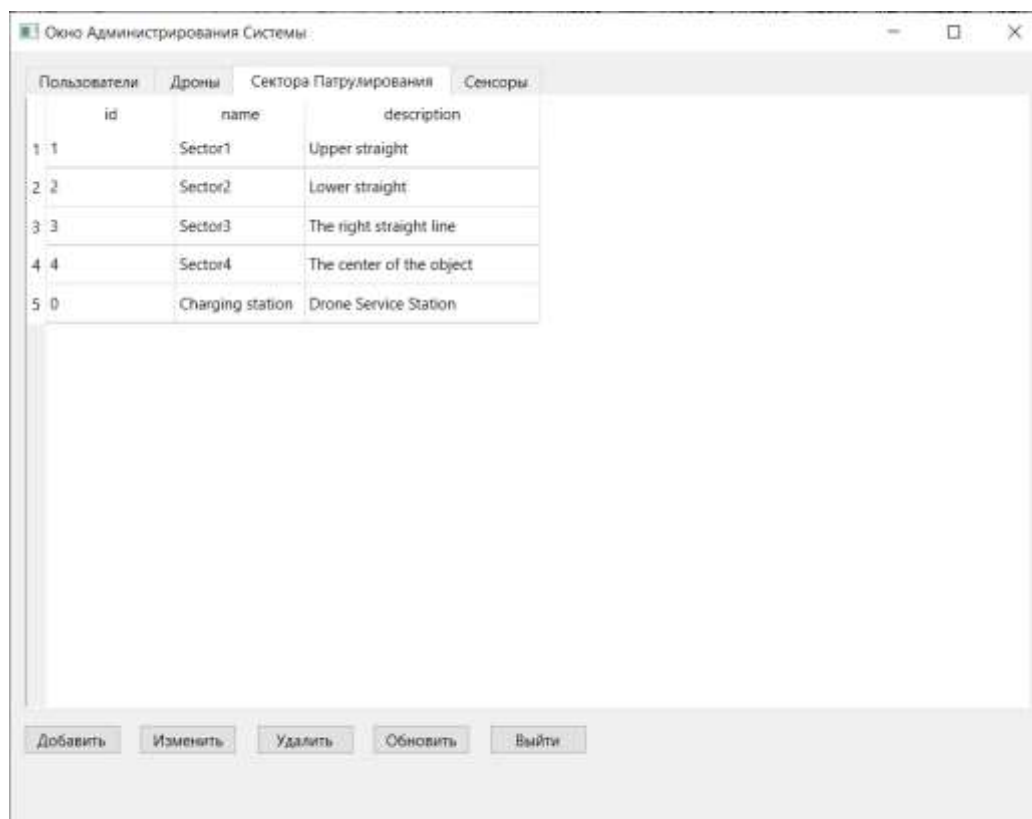


Рисунок 13 – окно с секторами и дронами

3.2.2 Добавление событий с дронов

Пользователь должен иметь возможность вручную добавлять данные о происшествиях, полученные дронами, в систему. На соответствующей странице предоставляется форма для селекции данных, где пользователь выбирает сектор, выбирает дрона и выбирает событие. После заполнения всех необходимых полей и нажатия кнопки "Отправить", данные сохраняются в базе данных.

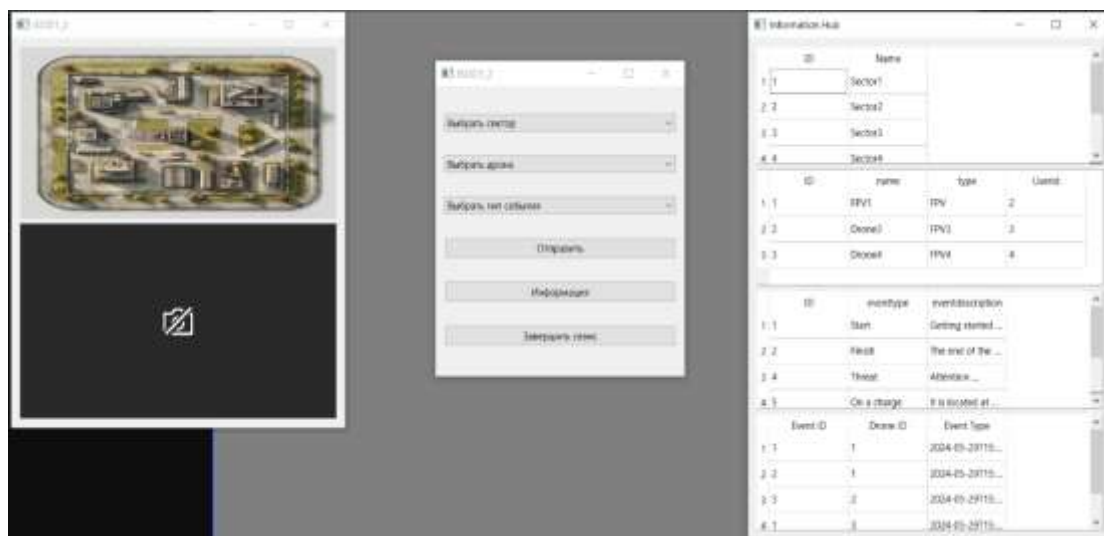


Рисунок 14 - Страница с добавлением данных с дронов

3.2.3 Анализ данных и формирования отчета с данными с дронов

Пользователь должен иметь возможность просматривать данные, собранные с дронов, в удобном формате и производить создание отчета. На странице должна отображаться таблица со всеми стратегическими данными. При нажатии на кнопку "Создание отчета" должно происходить выгрузка данных с форм.

The Analyst Window interface displays several data tables and a report generation button. The tables are as follows:

ID	Username	Role
1	KovalevD	Администратор
2	Vladimir	Оператор

ID	Name	Type	4
1	FPV1	FPV	2
2	Drone3	FPV3	3

ID	Name	Description
1	Sector1	Upper straight
2	Sector2	Lower straight

ID	Type	Description
1	TEST	TEST sensor
2	TT	TAA

DroneID	SectorID	TakeoffTimestamp	TimeEvent
---------	----------	------------------	-----------

EventID	DroneID	TimeEvent
1	1	2024-05-29T15:...
2	2	2024-05-29T15:...

Buttons: Обновить, Создание отчета

Рисунок 15 - Страница с просмотром анализа данных и выгрузки отчет

4. РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ

4.1 Базы данных

Информационная система для хранения и анализа данных с полей включает в себя базу данных PostgreSQL. База данных содержит следующие основные таблицы: `drones`, `patrolsectors`, `Users`, `sensors`, `event`, `DroneEvent`, `sectorDrone`, `dronesensors`. Каждая таблица имеет четко определенные поля и связи, что обеспечивает целостность и структурированность данных. Таблица `drones` хранит информацию о дронах, включая их уникальный идентификатор и название. Коллекция `patrolsectors` содержит данные о секторах, такие как название сектора и идентификатор соответствующего сектора. Таблица `Users` хранит данные о пользователях, их ролях и пароли. Таблица `sensors` содержит в себе данные о датчиках и их типах. Таблица `event` хранит данные о событиях и их краткое описание. Таблицы `DroneEvent`, `sectorDrone`, `dronesensors` содержат в себе поля с соответствующими идентификаторами, для соединения таблиц.

4.2 Приложение клиент

Декстопная часть приложения представляет собой интерфейс, реализованный с использованием фреймворка QT и C++. Интерфейс предоставляет средства для управления секторами и дронами данными, а также для анализа собранных данных.

Страница администрирования системы предоставляет возможность просматривать информацию о существующих секторах и дронах, добавлять новые сектора и дроны, редактировать данные существующих секторов и дронов и удалять ненужные записи. На странице должна быть представлена таблица с перечнем всех секторов и дронов. Для добавления нового сектора пользователь вводит название сектора и его описание, и нажимает кнопку "Добавить ". Для добавления нового дрона пользователь вводит название дрона, его тип и привязанного оператора, и нажимает кнопку "Добавить". Для

редактирования существующего поля пользователь выбирает нужное поле в таблице и нажимает кнопку "Редактировать", после чего открывается новая форма для ввода. Для удаления поля пользователь выбирает его в таблице и нажимает кнопку "Удалить", после чего система запрашивает подтверждение действия и, при согласии, удаляет запись из базы данных.

Страница Оператора дронов позволяет вручную добавлять данные о происшествиях, полученные дронами, в систему. На соответствующей странице предоставляется форма для селекции данных, где пользователь выбирает сектор, выбирает дрона и выбирает событие. После заполнения всех необходимых полей и нажатия кнопки "Отправить", данные сохраняются в базе данных.

Страница анализа данных предоставляет возможность просматривать данные, собранные с дронов, в удобном формате и производить создание отчета. На странице отображается таблица со всеми стратегическими данными. При нажатии на кнопку "Создание отчета" должно происходить выгрузка данных с форм.

ЗАКЛЮЧЕНИЕ

Таким образом, в ходе разработки информационной системы управления и анализа данных, охраны территорий было проведено исследование предметной области, в результате которого были поставлены цели и задачи информационной системы, выявлены основные пользователи системы. Было проведено проектирование и комплексное моделирование системы. Функциональное моделирование осуществлялось с использованием методологий IDEF0 и IDEF3, а также DFD. Помимо этого, использовалось объектное моделирование в методике UML. Моделирование позволило декомпонировать систему и выявить основные компоненты, а также сформировать структуру проекта. В ходе работы были сформированы технические требования, которые содержат информацию о функциональности, производительности, надежности, безопасности, интерфейсе и других аспектах разрабатываемого программного обеспечения.

Опираясь на эти требования, в итоге была реализована информационная система для хранения и анализа данных с полей. Система включает в себя десктопную часть обеспечивающую обмен данными между пользователями системы и базой данных PostgreSQL.

Ключевыми достоинствами разработанной информационной системы являются:

1. Автоматизация процессов: система позволяет сократить время выполнения операций и минимизировать ручной ввод данных благодаря автоматизированным процессам.
2. Улучшенная оперативность: благодаря возможности быстрого доступа к актуальным данным, пользователи могут оперативно реагировать на изменения и принимать решения на основе актуальной информации.

Общий результат проекта демонстрирует успешную реализацию информационной системы, способствующей повышению эффективности и качества управления охранными мероприятиями. Дальнейшее развитие системы может включать в себя добавление новых функциональных возможностей, улучшение пользовательского интерфейса и оптимизацию производительности для обеспечения более эффективной работы персонала.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Бланшет Саммерфилд. Qt4 Программирование GUI на C++. 2ed. – ДМК Пресс, 2008. – 150 с.
- 2 Шлее Макс. Профессиональное программирование на C++. +CD. Qt 4.8. – Москва: Вильямс, 2012. – 258 с.
- 3 Фейт, С. TCP / IP. Архитектура. Протоколы. Реализация / С. Фейт. – Москва: ИЛ, 2019. – 424 с.
- 4 Грекул, В. И. Проектирование информационных систем: учебное пособие / В. И. Грекул. – 2-е изд. – Москва: ИНТУИТ, 2016. – 570 с.
- 5 Андрей Боровский. Qt4.7+. Практическое программирование на C++ – Москва: ФОРУМ: ИНФРА-М, 2018. – 177 с.
- 6 Розенберг, Д. Применение объектного моделирования с использованием UML и анализ прецедентов: руководство / Д. Розенберг, К. Скотт. – Москва: ДМК Пресс, 2007. – 160 с
- 7 Марк Саммерфилд. Qt4.7+. Qt Профессиональное программирование (High tech) – Москва: ИЛ, 2014. – 347 с.

ПРИЛОЖЕНИЕ А

АВТОМАТИЗИРОВАННАЯ СИСТЕМА УПРАВЛЕНИЯ И АНАЛИЗА ДАННЫХ, ОХРАНЫ ТЕРРИТОРИЙ

Руководство программиста

Санкт-Петербург 2024

Оглавление

Назначение и условия применения программы	35
Характеристики программы	36
Вызов программы	37
Входные и выходные данные	37
Сообщения	38
Логика работы программы, функции и методы	39
Приложения.....	Ошибка! Закладка не определена.

Назначение и условия применения программы

Назначение программы:

Программа предназначена для управления процессами при охране территорий с использованием дронов и датчиков. Основная цель - обеспечение эффективных охранных мероприятий, взаимодействия операторов дронов, охраны и административным персоналом через автоматизированную систему управления секторами, датчиками, дронами, событиями, а также получение и обработка информации о секторах.

Основные функции программы:

1. Управление секторами, датчиками, дронами, событиями пользователями:

- Создание новых данных о секторах, датчиках, дронах, событиях и пользователях.
- Просмотр списков всех секторов, датчиков, дронов, событий и пользователей.
- Изменение действительной информации, конкретных о секторах, датчиках, дронах, событиях и пользователях.
- Удаление информации о секторах, датчиках, дронах, событиях и пользователях.

2. Работа с центром информации о секторах:

- Получение расширенной информации о выбранном секторе
- Просмотр всех элементов, принадлежащих сектору и подробной информации о них
- Получение текстовых описаний о секторе
- Генерация отчетов по сектору со всеми вытекающими элементами

3. Работа операторов дронов:

- Получение информации по выбранному сектору и дрону
- Получение видеопотока с работающего дрона
- Получение информации о сеансе подключения к дрону
- Непосредственная работа с дроном

4. **Возможности безопасности:**

- Настройка параметров системы, включая управление доступом пользователей к различным функциям.
- Возможность добавления и настройки ролей пользователей для разграничения доступа к информации и операциям в системе.

Условия применения:

- Программа устанавливается на ЭВМ машину, поддерживающую C++ , PostgreSQL, QT Creator что обеспечивает ее масштабируемость и высокую производительность.
- Для использования программы необходимо наличие вычислительных мощностей
- Доступ к программе контролируется системой авторизации и аутентификации, которая гарантирует защиту и конфиденциальность данных пользователей.
- Программа предоставляет различные роли для пользователей, что обеспечивает дополнительную безопасность и удобство при работе с системой.

Характеристики программы

Каждое рабочее место должно быть оборудовано компьютером с следующими техническими характеристиками: процессор Intel Core i7-9700, 12 ГБ оперативной памяти, 1 ТБ жесткого диска и т.д. Компьютер должен быть подключен к локальной сети и интернету через Wi-Fi роутер Asus RT-AC86U. В качестве операционной системы используется Windows 10. На компьютерах

должна быть установлена среда выполнения C++ и среда работы с базами данных PostgreSQL.

Временные характеристики: Обработка данных происходит в режиме реального времени, реакция на действия пользователя является почти мгновенной. **Временной ресурс:** Отклик на действия 1-2 секунд.

Режим работы: В зависимости от варианта установки, может быть доступна круглосуточно или в рабочие часы предприятия.

Вызов программы

Процедуры вызова: Доступ к программному обеспечению осуществляется через интерфейс. Пользователи могут выполнять такие операции, как CRUD-операции для всех сущностей или доступ к информационному центру полей и странице оператора дрона, через формы, которые внутренне вызывают соответствующие методы обслуживания.

Методы передачи параметров управления и данных: данные передаются через connect-запросы, обрабатывая методы для получения и передачи данных.

Входные и выходные данные

Организация используемой входной информации:

Входная информация может быть представлена в виде:

- Ввода с клавиатуры
- Использованием оптического манипулятора вида «Мышь»
- Получение видеоданных с дронов и предназначенных устройств

Выходные данные:

Выходная информация может быть реализована в виде:

- Текстовых данных
- Обработанных видеопотоков дронов
- Передачи данных в информационный центр системы

Кодировка: для текстовых данных применяется кодировка UTF-8, которая обеспечивает совместимость и предотвращает проблемы с кодировкой.

Сообщения

Обработка ошибок на форме:

- Поля формы проверяются на соответствие требованиям (например, числовые данные на соответствие действительности, создаваемые сущности на предмет дубликатов, обязательные поля).

Пользовательский интерфейс для ошибок:

- Ошибки на форме отображаются поверх полей ввода.
- В случае ошибок валидации поверх полей ввода выводится сообщение об ошибке, в которой говорится как ее исправить.

Примеры сообщений об ошибках на форме login средствами QMessageBox:

"Ошибка авторизации", "Неверное имя пользователя или пароль."

Данная ошибка следует из нарушения логики заполнения полей, подробнее можно увидеть на ui форме.

Обработка ошибок в C++

Обработка ошибок в классах

Логика обработки:

- Классы отвечают за реализацию логики и взаимодействуют с базой данных или другими классами. При возникновении ошибок, таких как нарушение правил или проблемы доступа к ресурсам, генерируются исключения.

- Возникшие исключения регистрируются в консоли разработчика.

Механизмы восстановления после ошибок:

- В случае обнаружения ошибки классы могут пытаться выполнить альтернативные действия, возвращать стандартные значения.

Примеры сообщений об ошибках в сервисах с выводом в консоль разработчика:

- "Ошибка доступа к базе данных. Попробуйте повторить операцию позже."
- "Не удалось найти запрашиваемый ресурс. Возможно, он был удалён."

Примеры Push уведомлений на ошибки:

"Ошибка выполнения запроса:"

"Ошибка", "Поле должно содержать только буквы и цифры."

Логика работы программы, функции и методы

Программа состоит из 3 видов файлов:

1. Заголовочные файлы(.h) - Файлы с расширением .h, также известные как заголовочные файлы (header files), используются в языке программирования С и С++ для объявления функций, переменных и констант. Они содержат прототипы функций и объявления структур данных, которые будут использоваться в программе. Обычно .h файлы содержат только объявления (declaration) без определений (definition) функций или переменных.
2. Файлы с исходным кодом (.cpp) - В файлах .cpp содержится реализация функций, классов, переменных и других конструкций, определенных в программе. То есть, в отличие от заголовочных файлов (.h), которые содержат только объявления, .cpp файлы содержат их реализации.
3. Файлы с расширением .ui (.ui) - это файлы пользовательского интерфейса, которые используются в различных инструментах для разработки графических пользовательских интерфейсов (GUI). Одним из самых распространенных инструментов для создания .ui файлов является Qt Designer, который является частью фреймворка Qt для разработки программного обеспечения.

Заголовочные файлы(.h)

Все заголовочные файлы похожи между собой, для примера рассмотрим файл `login.h` отвечающий за объявления класса, библиотек и методов участвующих в процессе авторизации.

Файл `login.h` - это заголовочный файл (header file), который содержит объявление класса `Login` (Листинг файла представлен в [приложении 1](#)). Класс `Login` наследует функциональность классов `QMainWindow` и `Ui::Login` и дополняет их собственными методами и слотами.

Директива `#ifndef LOGIN_H` начинает стражу включения (include guard), которая предотвращает множественное включение файла. Если файл `login.h` уже был включен в программу, то его содержимое не будет включено повторно.

Директива `#define LOGIN_H` определяет метку `LOGIN_H`, которая используется в страже включения.

Затем следует включение необходимых заголовочных файлов, в том числе `QMainWindow` и `ui_login.h`.

Далее идет объявление класса `Login`, который наследует от `QMainWindow` и `Ui::Login`. Он содержит открытые, закрытые и защищенные члены класса, включая конструкторы, деструктор, слоты и приватные методы.

Секция `private slots:` содержит объявление слотов, которые реагируют на сигналы (обычно от пользовательского интерфейса).

Секция `private:` содержит приватные члены класса, включая указатель `ui`, который представляет объект пользовательского интерфейса, и приватные методы, такие как `openUserWindow()`, `openAdminWindow()` и `openAnalystWindow()`.

Стража включения `#endif // LOGIN_H` завершает страж включения.

Файлы с исходным кодом (.cpp)

Все файлы с исходным кодом похожи между собой, для примера рассмотрим файл `login.cpp` отвечающий за реализацию методов класса авторизации.

Файл `login.cpp` содержит реализацию методов класса `Login`. Этот класс представляет окно входа в систему. Он отвечает за обработку событий, связанных с входом пользователя, а также за открытие различных окон приложения в зависимости от роли пользователя.

Метод `handleLoginButtonClicked()` вызывается при нажатии на кнопку входа. Он получает введенное пользователем имя и пароль, а затем проверяет их с помощью метода `authenticateUser()` класса `DatabaseManager`. В зависимости от роли пользователя окно входа открывает соответствующее окно приложения.

Методы `openUserWindow()`, `openAnalystWindow()` и `openAdminWindow()` отвечают за открытие окон для пользователя, аналитика и администратора соответственно.

Файлы с расширением .ui

Все файлы с расширением `.ui` похожи между собой структурой, для примера рассмотрим файл `login.ui` отвечающий за создание пользовательского интерфейса класса авторизации.

Файл `login.ui` представляет собой XML-файл, который описывает пользовательский интерфейс для окна входа в систему. Он создан с помощью инструмента `Qt Designer` или другого редактора форм.

Файл содержит различные элементы интерфейса, такие как окно приложения, метки, поля ввода и кнопки. Эти элементы размещены с помощью различных макетов (layout), таких как `QVBoxLayout`.

Каждый элемент имеет свои свойства, такие как текст, геометрия, выравнивание и т.д. Например, метки "Логин" и "Пароль" выровнены по левому краю, а поля ввода - по центру.

Сигналы и слоты также определены в файле, чтобы обеспечить связь между элементами интерфейса и функциональностью приложения. Например, сигнал `clicked()` кнопки "Войти" соединен со слотом `on_loginButton_clicked()`, который обрабатывает событие нажатия кнопки.

Особые классы:

“DatabaseManager”): Этот класс отвечает за взаимодействие с базой данных. Он реализует методы для открытия и закрытия соединения с базой данных, аутентификации пользователей, создания таблиц, добавления данных и извлечения информации из базы данных.

Конфигурация безопасности:

Код класса “DatabaseManager” (Листинг файла представлен в приложении 4). представляет собой конфигурацию безопасности для взаимодействия с базой данных в приложении. Он обеспечивает безопасное хранение паролей пользователей, так как использует хэширование паролей перед их сохранением и сравнением при аутентификации.

Код состоит из нескольких основных частей:

1. Открытие и закрытие базы данных: Класс обеспечивает безопасное открытие и закрытие соединения с базой данных.

2. Аутентификация пользователей: Метод `authenticateUser` выполняет аутентификацию пользователей, проверяя соответствие имени пользователя и хэшированного пароля в базе данных.

3. Создание таблиц: Метод `createTables` создает несколько таблиц в базе данных для хранения информации о пользователях, дронах, секторах патрулирования, сенсорах и событиях.

4. Добавление данных: Класс предоставляет методы для добавления пользователей, дронов, секторов патрулирования, сенсоров и событий в базу данных.

5. Извлечение данных: Класс также реализует методы для извлечения информации о секторах патрулирования, дронах, сенсорах и событиях из базы данных.

Этот файл необходим для обеспечения безопасности приложения, а так же обеспечения работы с базой данных. Он позволяет контролировать доступ к различным частям приложения в зависимости от роли пользователя.

ПРИЛОЖЕНИЕ Б

ИНФОРМАЦИОННАЯ СИСТЕМА ДЛЯ ХРАНЕНИЯ И АНАЛИЗА ДАННЫХ С ПОЛЕЙ

Руководство пользователя

Санкт-Петербург 2024

Оглавление

1. Введение.....	45
1.1 Область применения.....	45
1.2 Описание возможностей.....	45
1.3 Обучение пользователя.....	45
2. Назначение системы и условия пользования	46
3. Мероприятия проводящиеся перед работой	47
3.1 Установка необходимых программ и сопутствующих данных	47
3.2 Алгоритм проверки работоспособности	47
4. Функционал приложения.....	48
5. Аварийные ситуации	53

1. Введение

1.1 Область применения

Требования настоящего документа применяются при проведении всех видов тестирований (предварительное, пред эксплуатационная, эксплуатационная) и непосредственно при самой эксплуатации.

1.2 Описание возможностей

Система ИСУ-Б-ОТ-2023 предназначена для использования компанией «Спецзащита», и используется как информационная система для работы с информацией в данном предприятии. Пользователи данной системы (сотрудники данной компании) должны использовать данную программу для работы с информацией о секторах, сенсорах, дронах, событиях и сотрудниках, для такого взаимодействия программа представляет данный функционал:

1. Добавление информации о секторах, сенсорах, дронах, событиях и сотрудниках.
2. Удаление и редактирование информации о секторах, сенсорах, дронах, событиях и сотрудниках
3. Просмотр данной информации
4. Получение видео информации о событиях в охраняемых секторах
5. Генерации отчетов по определенному сектору
6. Ведение и запись статистических данных и журнала событий

В целях безопасности программа снабжена защищённым доступом к системе благодаря наличию авторизации и хэш-паролей

1.3 Обучение пользователя

Пользователь, работающий в данной системе, должен обладать следующими навыками и знаниями:

1. Опыт работы с операционной системой MS Windows, версиями не ниже 7 (7, 8, 8.1, 10, 11) или Linux любых стандартных сборок (Debian, Ubuntu, Astra Linux)
2. Уметь выполнять базовые операции, такие как установка и удаление программ, работа с файловой системой, настройка параметров системы.
3. Знание соответствующей предметной области, связанной с программным обеспечением для управления беспилотными летательными аппаратами (дронами).
4. Понимание задач, связанных с мониторингом, патрулированием и управлением дронами, а также знаний в области безопасности и эффективного использования этих устройств.
5. Пользователь должен иметь опыт работы со страничными формами заполнения, включающими взаимодействие с элементами управления, такими как текстовые поля, выпадающие списки, кнопки и таблицы.
6. Основы работы с базами данных
7. Опыт работы с Десктопными программами

2. Назначение системы и условия пользования

Эта система разработана для упрощения взаимодействия сотрудников с необходимой информацией и повышении безопасности при работе по охране территорий, а также для структурирования данных с целью повышения эффективности охранных мероприятий. Предоставлены удобные и эффективные функции для отслеживания секторов и связанных с ними событиями, дронами (п.1.2). Программа должна эффективно работать на операционных системах (п.1.3). Система постоянно доступна трем пользователям с разными правами доступа для выполнения работы, а именно: Администратор, Оператор дронов, Аналитик.

3. Мероприятия проводящиеся перед работой

3.1 Установка необходимых программ и сопутствующих данных

Для запуска предоставленной системы необходимо установить на служебном компьютере базу данных Postgresql, версия которой должна быть 15 или выше, данная операция повторяется со всеми компьютерами, вовлеченными в работу системы. Далее необходимо разместить программу в памяти операционной системы(п.1.3). После развертывания программа будет доступна только в локальном режиме, без выхода в интернет. После первого включения система создаст персональные данные для входа в систему, они будут предоставлены администратору в ходе приемки проекта.

3.2 Алгоритм проверки работоспособности

После проведения всех мероприятий по установке системы необходимо проверить ее работу индивидуально на каждом рабочем месте:

1. Запустите программу в одной из операционных систем присутствующих в списке, представленном в п. 1.3. Для этого необходимо нажать на ярлык программы и дождаться ее запуска
2. В открывшемся окне входа в систему ввести предоставленные ранее данные (логин и пароль), после чего нажать кнопку «Войти»
3. При успешном входе во вкладке появится ролевая страница для взаимодействия с системой
4. При необходимости проверить непосредственно работу системы на конкретных функциях

В случае возникновения проблем на каком-либо этапе проверки работоспособности необходимо обратиться к администратору системы.

4. Функционал приложения

Пользователям приложения предоставляются следующие функции для взаимодействия с информационной системой (функционал распределен по ролям пользователей):

Роль «Оператор дронов»:

Для операторов дронов открыт доступ к «Окну оператора». Данная страница позволяет выбрать дрон из всех доступных для просмотра данных о нем включая связанные с ним данные.

После выбора становится доступна функция выбора одного из секторов принадлежащих этому дрону для просмотра информации о нем.

Параллельно с этим происходит сеанс работы с дроном и вывод изображения с его камеры.

Так же постоянно доступной является кнопка закрытия ролевого окна, при нажатии на которую пользователь закроет программу и кнопка вывода информации о секторах, дронах и подключенных режимах работы сенсоров.

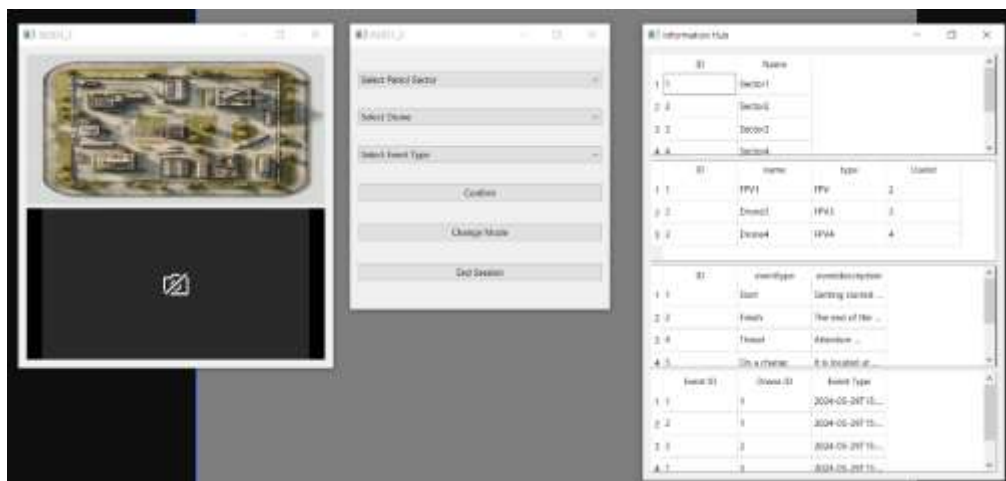


Рис. 1 – внешний вид окна «Страница оператора»

Роль «Аналитик»:

Для пользователей с ролью «Аналитик» предоставлен доступ к странице «Аналитическое окно». Данная страница позволяет увидеть все важнейшие статистические данные: Всех пользователей системы и их данные, всех доступных дронов их данные, все сектора и их данные, все сенсоры и их данные и аналитические данные полученные во время патруля.

Функция обновления данных. Данная функция начинает работу при нажатии на кнопку «Refresh», после чего получает данные и обновляет текущие значения характеристик.

Функция генерации отчета. Данная функция начинает работу при нажатии на кнопку «Export Report», после чего перенаправляет пользователя в файловый редактор ЭВМ, для выбора папки сохранения данных. После чего сохраняет данные в файл формата .txt. Возврат на предыдущую страницу происходит автоматически.

Так же постоянно доступна кнопка возврата на главный экран, нажатие на которую перенаправит пользователя на искомую страницу.

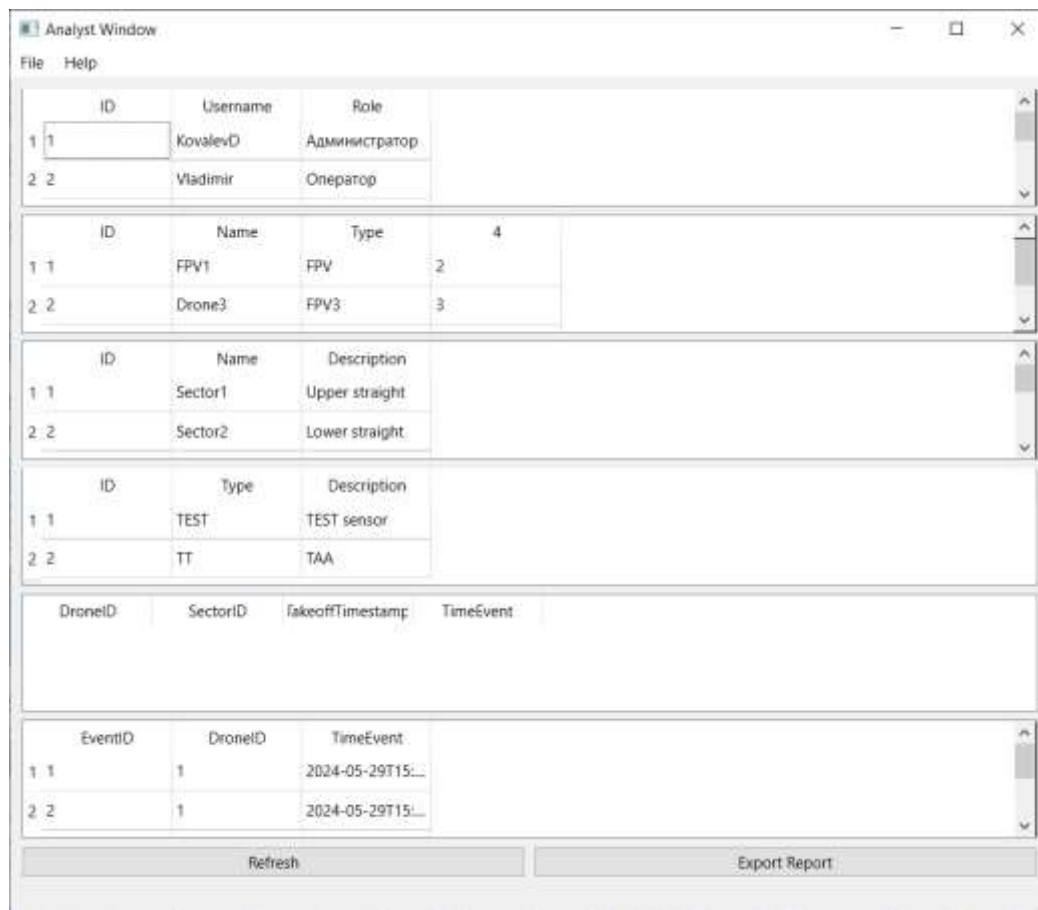


Рис. 2 – внешний вид страницы «Аналитическое окно»

Роль «Администратор»:

Для пользователя с ролью «Администратор» предоставляется доступ к окну «Окно Администрирования системы».

Окно просмотра списка пользователей\дронов\секторов\девайсов предоставляют список для их просмотра, добавления, редактирования и удаления. Во всех трех списках есть функционал для добавления нового элемента, удаление элемента, редактирования элемента. Разница состоит лишь в содержании списков и в формах для добавления и редактирования. Перенаправление на другую страницу предусмотрено только для функций добавления и редактирования данных. На всех представленных формах имеется кнопка для их закрытия/отмены.

Страница просмотра списка полей практически идентична вышеописанным, за исключением добавленной функции обновления данных и возможности присвоить дрон\растение\удобрение к выбранному полю. Функция обновления данных происходит автоматически при модификации данных.

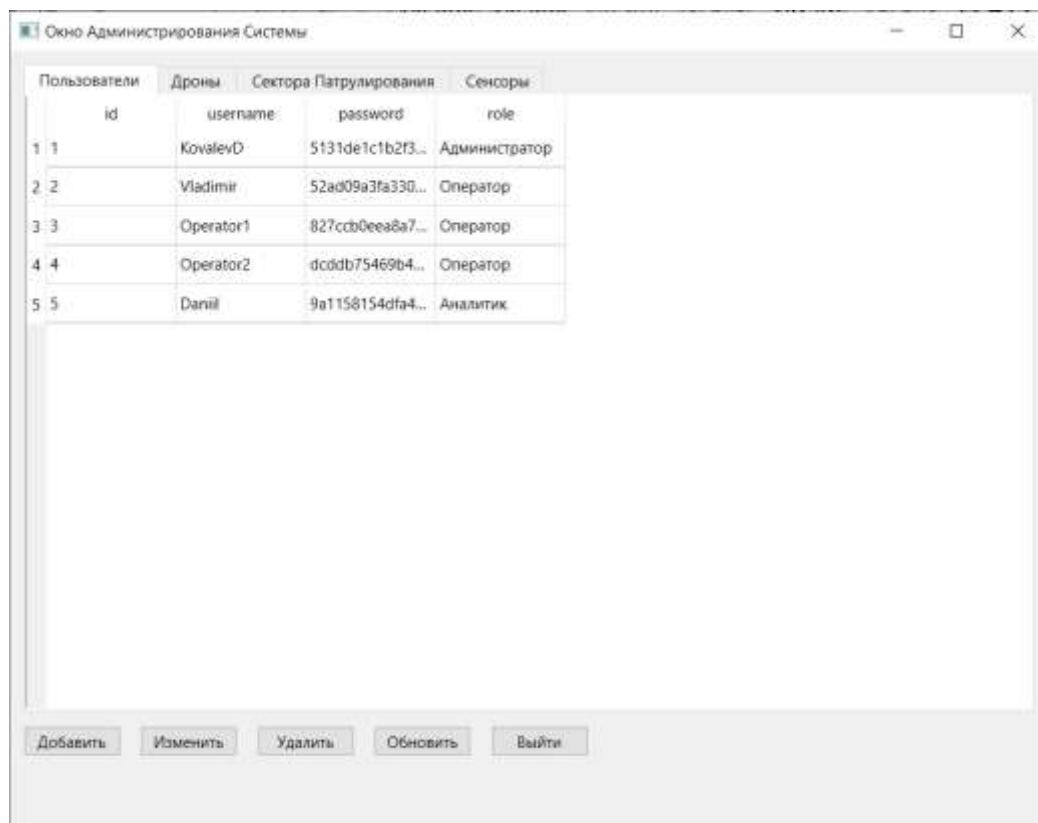


Рис. 3 – общий вид списков элементов пользователи

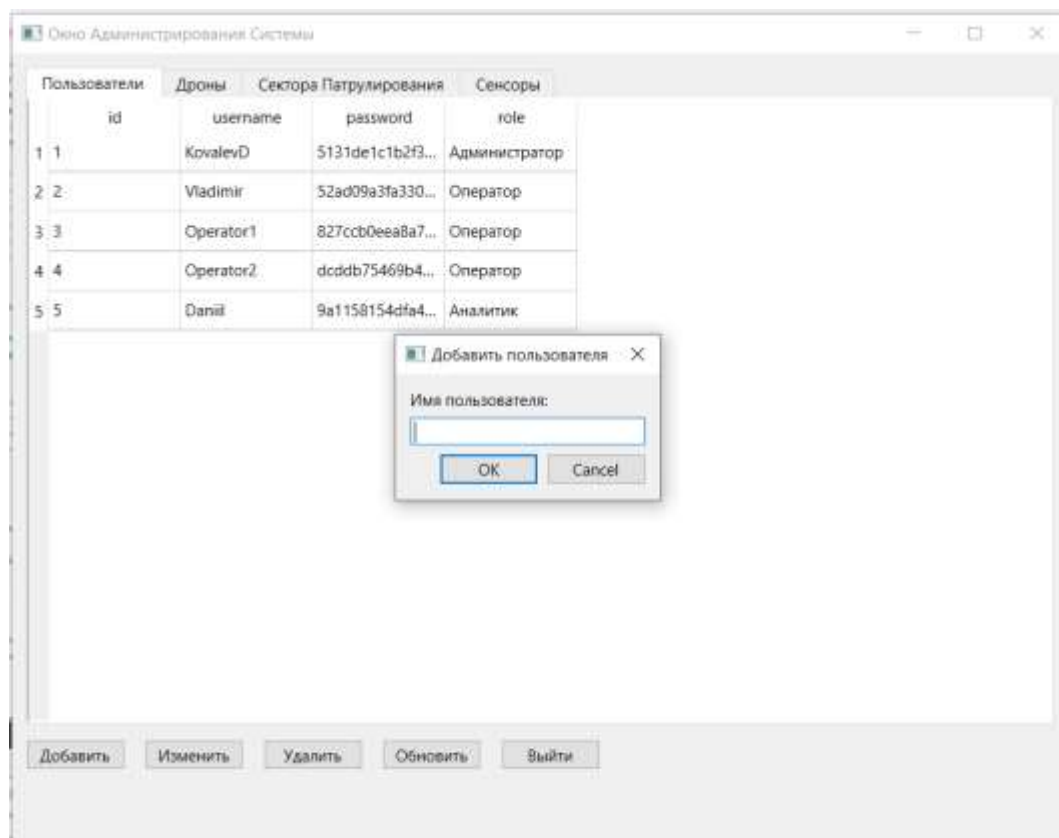


Рис. 4 – общий вид формы добавления элемента

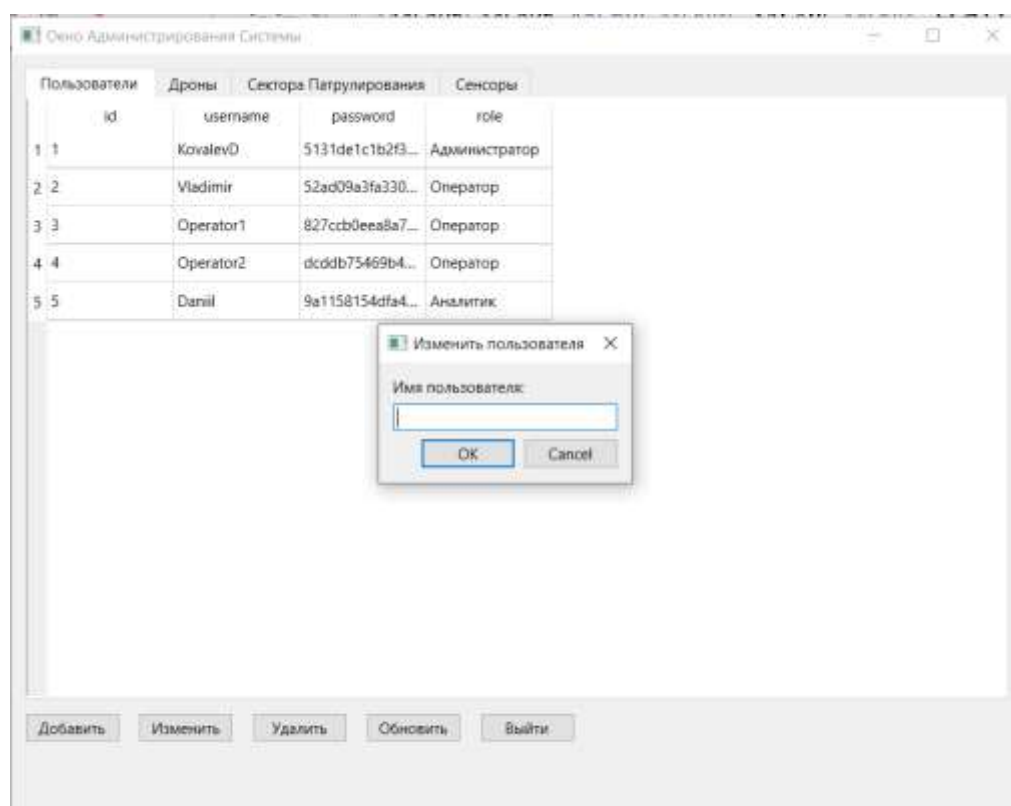


Рис. 5 – общий вид формы редактирования элемента

Для всех ролей:

Независимо от роли доступ к окну авторизации предоставлен всем пользователям. Окно авторизации реализует функцию навигации по информационной системе и функцию входа пользователей.

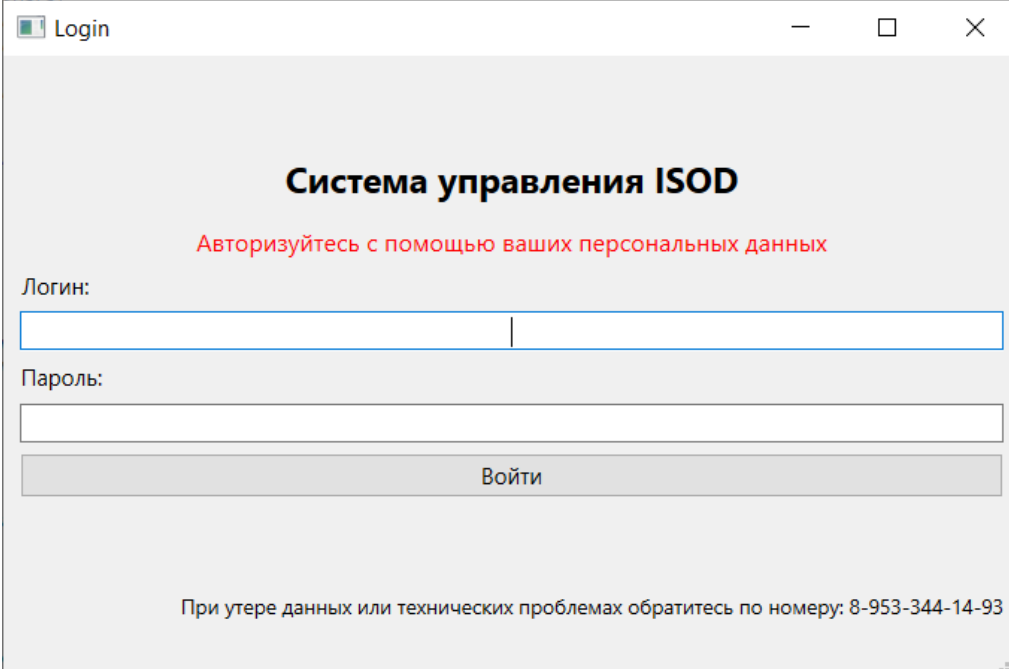


Рис. 6 – вид главной страницы, она же авторизация

5. Аварийные ситуации

Класс ошибки	Ошибка	Описание ошибки	Требуемые действия пользователя при возникновении ошибки
Сбой информационной системы	Невозможно отобразить окно	Проблемы с доступом к информационной системе.	Обратится к персоналу технической поддержки,

			администратору
	Ошибка: отсутствие доступа	Пользователь не авторизован в системе	Обратится к администратору для получения подробной информации
	Ошибка: Сбой аутентификаци и. Повторите попытку	Неверно введено имя пользователя или пароль, либо такая учетная запись не зарегистрирова на.	Нужно повторить ввод имени пользователя и пароля. Если возникают проблемы обратится к администратору
	Сбой электропитани я	Нет электропитани я произошел сбой в электропитани и.	Рабочий компьютер выключился или перезагрузился
			Перезагрузить рабочий компьютер. Проверить доступность информационной системы, если все работает корректно, то проверить корректно ли сохранены данные, если не работает, обратится в поддержку или

			администратору
Сбой периферийных устройств	Нет локального взаимодействи я блока компьютера с периферийным и устройствами	Отсутствует возможность продолжить работу с информационно й системой. Нет устройств ввода/вывода информации	Перезагрузить рабочий компьютер. Проверить доступность периферийных устройств, если все работает корректно, то продолжить работу, если не работает, обратится в поддержку или к администратору