ПРИЛОЖЕНИЕ Д. БИЗНЕС ПРОЦЕССЫ

АННОТАЦИЯ

В данном программном документе представлены бизнес-процессы, структурированный следующим образом: модель AS-IS, модель TO-BE.

Модель AS-IS: содержит модель IDEF0 AS-IS, до введения программного комплекса, бизнес-процесса и его описание, а также диаграмму прецедентов до внедрения программного комплекса;

Модель TO-BE: содержит модель IDEF0 TO-BE, после введения программного комплекса, бизнес-процесса и его описание, а также диаграмму прецедентов после внедрения программного комплекса;

СОДЕРЖАНИЕ

[1. БИЗНЕС ПРОЦЕССЫ 3](#_Toc198132389)

[1.1. Модель AS-IS 3](#_Toc198132390)

[1.2. Диаграмма прецедентов после внедрения ИС и модель TO-BE 7](#_Toc198132391)

1. БИЗНЕС ПРОЦЕССЫ
   1. Модель AS-IS

На рисунке 1 указан бизнес-процесс «Отслеживания устройств» модели A-0.

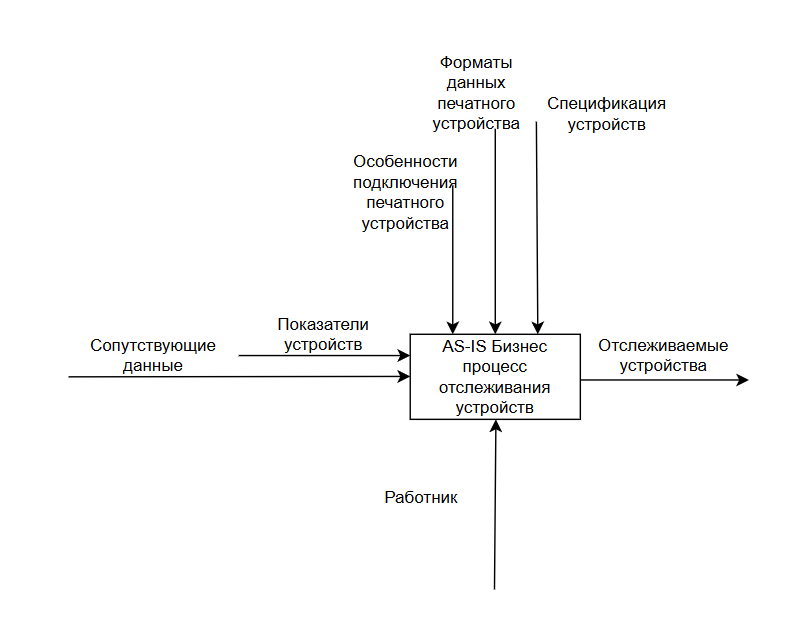


Рисунок 1 – Отслеживание устройств

У представленной модели A0 содержатся этапы от добавления устройства до подключения счетчика.

В модель передаются данные:

* Показатели устройств: физические показатели устройств, считываемые каким-либо образом, для дальнейшего внесения в БД.
* Сопутствующие данные: расположение устройства и наличие аварийного состояния у него.чё

Данные на выход модели:

* Отслеживаемые устройства

Механизмы модели: работник (заполняет записи).

На рисунке 2 продемонстрирована модель А1 этапа добавления устройства.

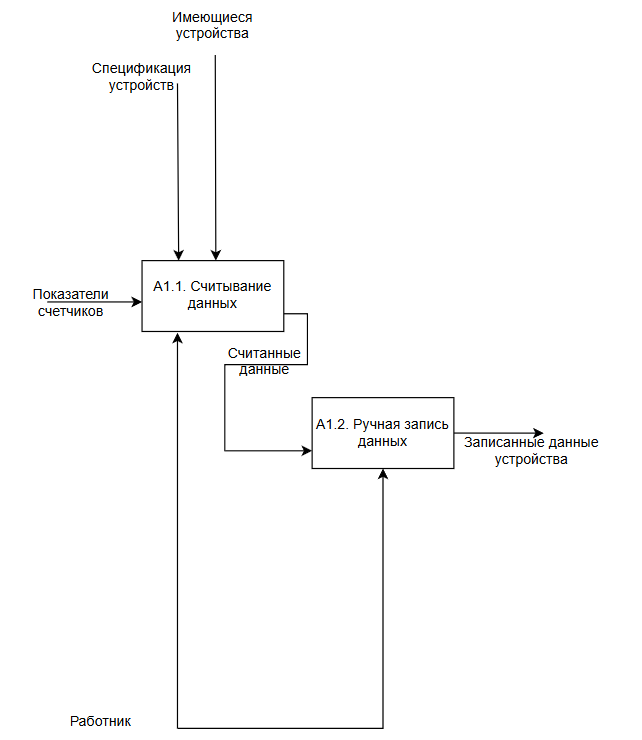


Рисунок 2 – Этап А1 Добавление устройства

На рисунке 3 продемонстрирована модель А2 этапа изменения устройства.

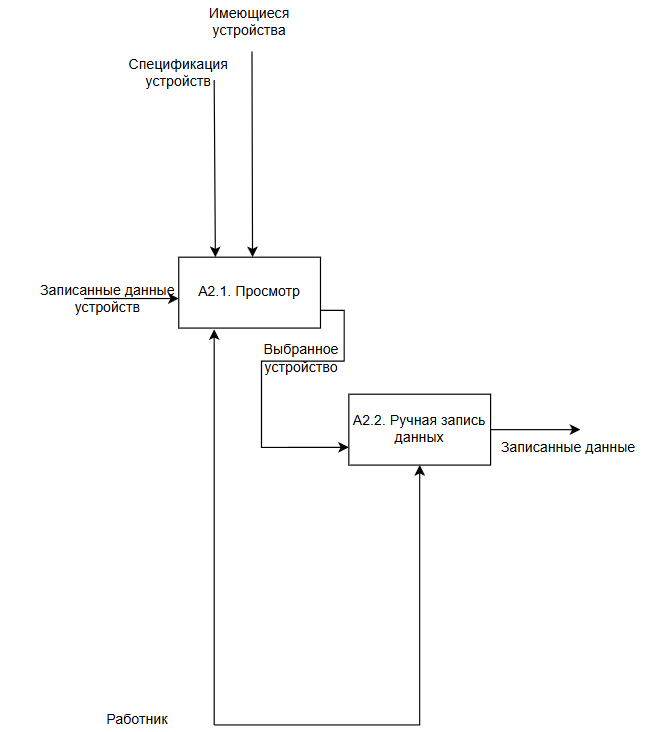


Рисунок 3 – Этап А2 Изменение устройства

На рисунке 4 продемонстрирована модель А3 этапа подключения счетчика.

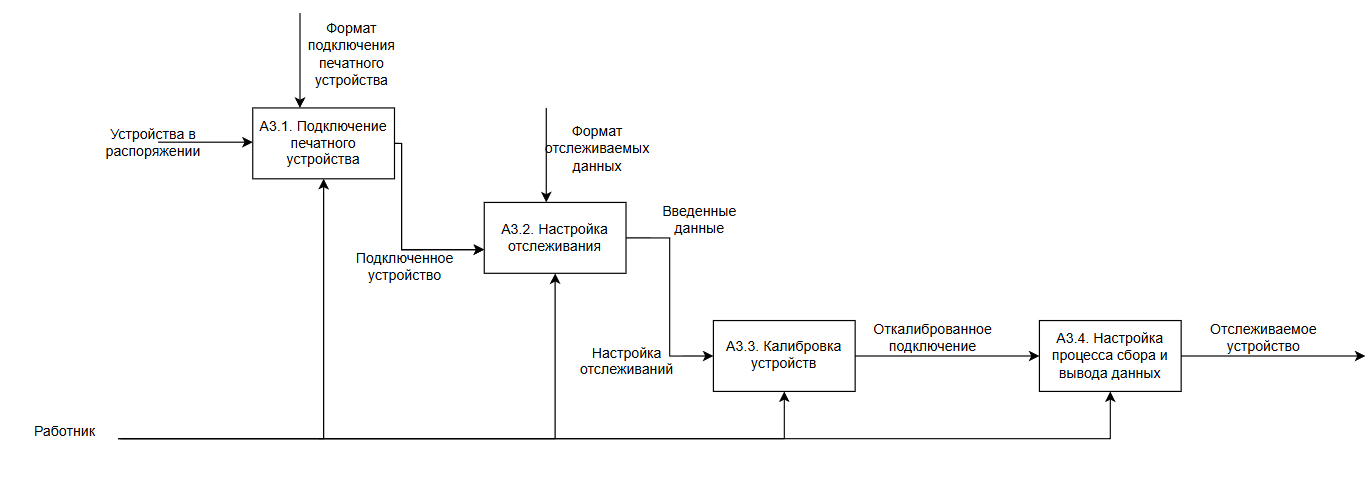


Рисунок 4 – Этап А3 Подключение счетчика

Проблемы бизнес-процесса до внедрения

1. Ручное ведение записей:
   * Использование бумажных носителей для записи данных устарело и неэффективно. Это может привести к ошибкам, потере данных и задержкам в обработке информации.
2. Отсутствие автоматизации:
   * Процесс не автоматизирован, что увеличивает время на выполнение задач и снижает общую производительность. Современные системы автоматизации могли бы значительно ускорить процесс и уменьшить количество ошибок.
3. Недостаток интеграции:
   * Отсутствие интеграции между различными этапами процесса может привести к дублированию данных и необходимости многократного ввода одной и той же информации, что увеличивает риск ошибок.

Критические ошибки алгоритма:

1. Отсутствие централизованного хранения данных:
   * Данные хранятся на бумажных носителях, что затрудняет их поиск и использование. Централизованная база данных могла бы обеспечить быстрый доступ к информации и её защиту.
2. Высокий риск потери данных:
   * Бумажные записи могут быть легко утеряны или повреждены, что приведет к потере важной информации. Электронное хранение данных с резервным копированием могло бы минимизировать этот риск.
3. Невозможность быстрого анализа данных:
   * Вручную записанные данные сложно и долго анализировать. Современные программные решения позволяют быстро обрабатывать и анализировать данные, что способствует принятию обоснованных решений.

Решение проблем данного бизнес-процесса:

1. Автоматизация сбора данных:
   * Внедрение электронных форм для сбора данных вместо бумажных носителей. Это позволит автоматически собирать и обрабатывать данные, уменьшая вероятность ошибок и ускоряя процесс.
2. Централизованное хранение данных:
   * Использование базы данных для хранения всех собранных данных. Это обеспечит быстрый доступ к информации, её защиту и возможность анализа.
3. Интеграция процессов:
   * Интеграция различных этапов процесса через единую информационную систему. Это позволит избежать дублирования данных и улучшит взаимодействие между различными участниками процесса.
   1. Диаграмма прецедентов после внедрения ИС и модель TO-BE

Модель AS-IS строится на основе записей показателей на бумажный носитель, а модель ТО-ВЕ на основе добавление устройств в приложение.

Эти изменения приведут к значительному сокращению времени на поиск и отсеивание на профессиональных фотографов, а также отсеивание по навыкам и месту работы.

Проект представляет из себя программный комплекс, состоящий из двух Desktop-приложений на фреймворке QT C++ и БД PostgreSQL и одного мобильного приложения на языке Java для ОС Android.

В приложении должно быть две основные роли: пользователь и администратор.

У пользователя и администратора две разных части приложения, серверная и клиентская. Серверная часть это Desktop-приложение, запускающее сервер и предоставляющая возможность пользоваться админ-панелью администратору после авторизации с соответствующей ролью. Клиентская часть – это два приложения с одним и тем же функционалом, одно из которых Desktop, другое – для Android.

У админа есть отдельная админ-панель, с необходимыми функциями, такими как CRUD-операции расположений, устройств и пользователей.

На рисунке 5 приведена контекстная диаграмма для модели бизнес-процессов деятельности приложения по размещению и отслеживанию. Контекстная диаграмма представляет собой самое общее описание системы и ее взаимодействие внутри себя.

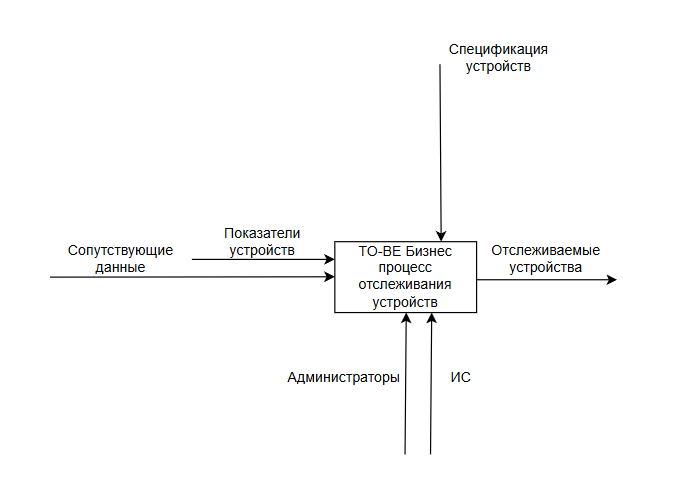


Рисунок 5 – Контекстная диаграмма

Для подробного рассмотрения работы приложения необходимо провести процесс декомпозиции контекстной диаграммы на рисунке 6.

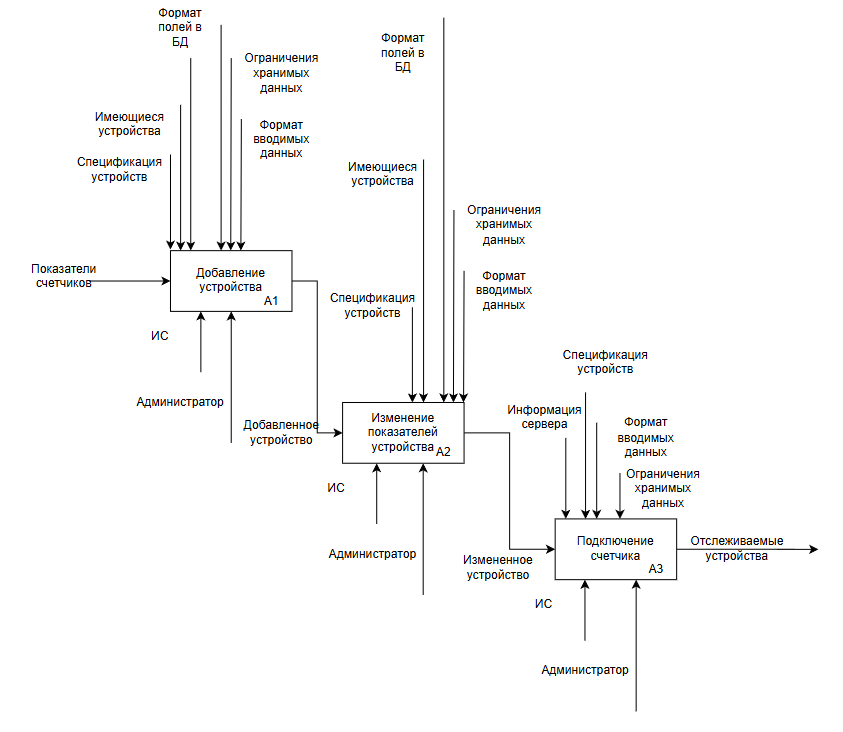


Рисунок 6 – Диаграмма декомпозиции

Ниже на рисунках 7-9 представлены модели IDEF0 для приложения контроля аварийных ситуаций устройств.

Ниже указан бизнес-процесс «Добавление устройства» модели A-1. Выходом блока «Добавление устройства» является добавленное устройство в БД в процессе выполнения данного бизнес-процесса.

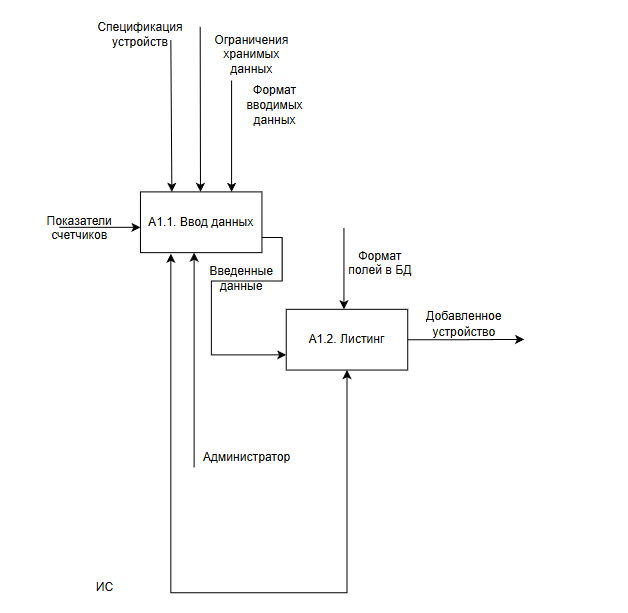


Рисунок 7 – Модель А1, добавление устройства

На рисунке 8 представлена модель А2, которая отвечает за изменение показателей устройства.

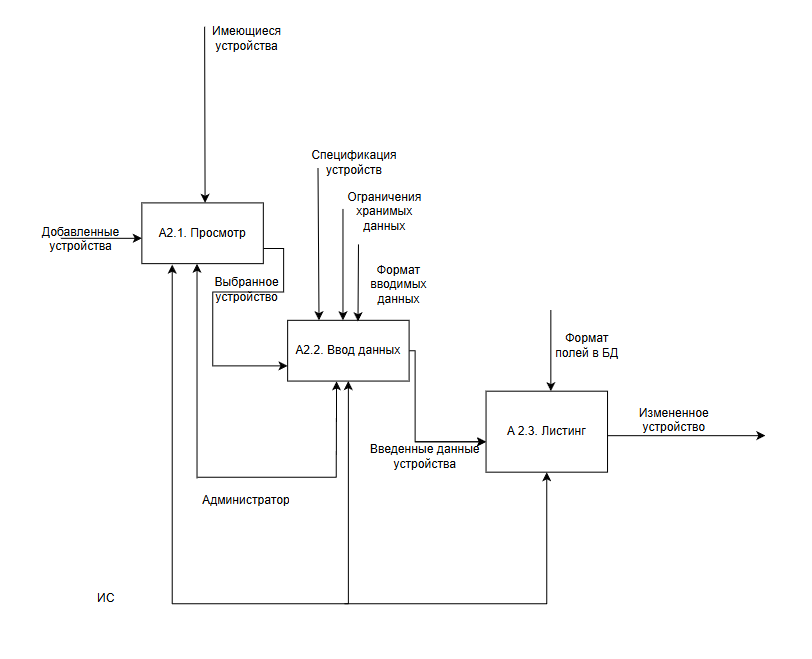


Рисунок 8 – Модель А2, изменение устройства

На рисунке 9 представлена модель А3, которая отвечает за подключение счетчика устройств.

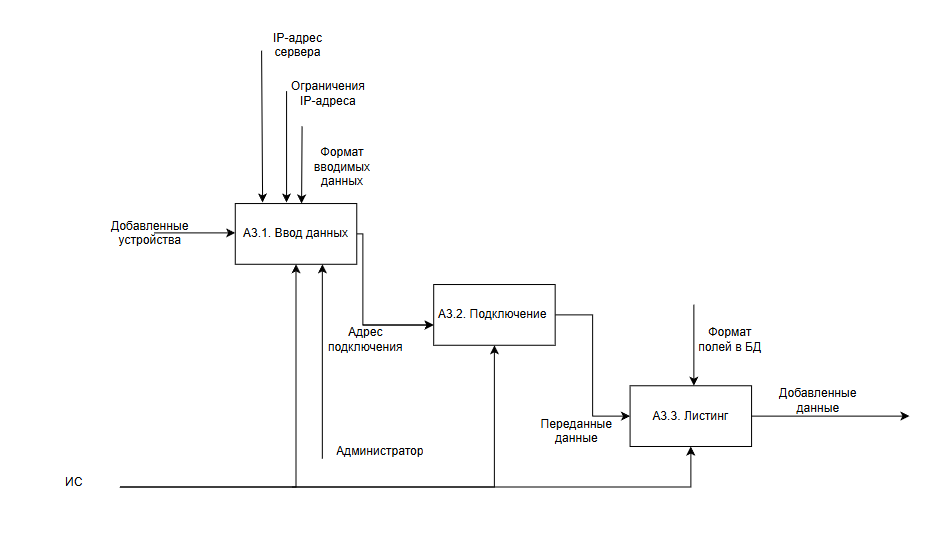


Рисунок 9 – Модель А3, поэтапное приготовление рецепта

На рисунке 10 представлена диаграмма прецедентов для приложения контроля аварийных ситуаций устройств.

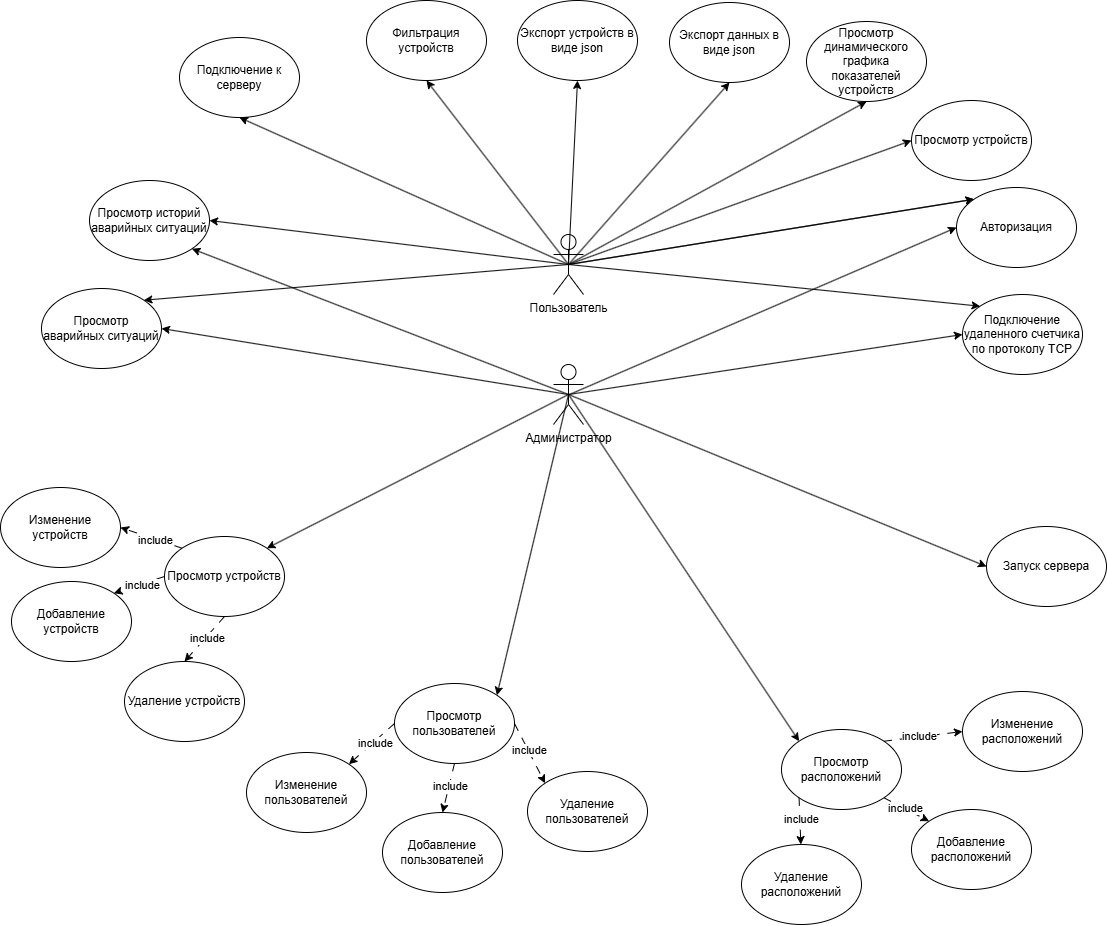


Рисунок 10 – Диаграмма прецедентов

На рисунке 11 представлен алгоритм авторизации внутри программы.

В алгоритме «Авторизация» пользователь открывает окно авторизации, вводит данные для авторизации, далее данные передаются в логику приложения, где она сравнивается с имеющимися данными из БД, и возвращает результат: найден ли пользователь с таким логином и паролем или же нет. Если результат не удовлетворителен, то пользователя переносит обратно на этап ввода данных для авторизации. Если же пользователь найден, то для начала проверяется наличие у него подходящей роли. Если роль не подходит, то пользователя возвращает на этап ввода данных для авторизации, если подходит, то пользователь авторизуется и переходит в основную часть приложения.

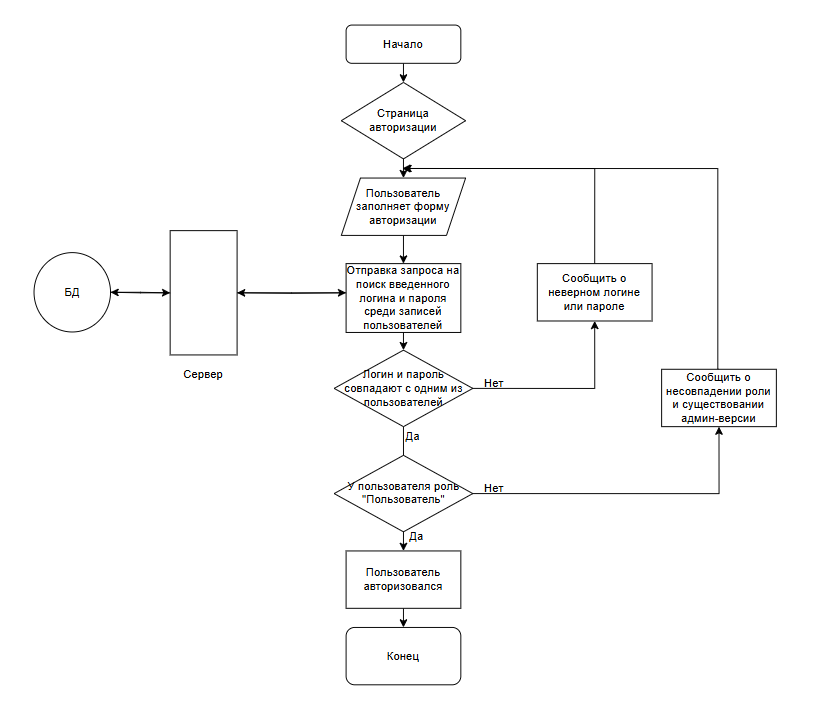


Рисунок 11 – Алгоритм авторизации