Государственное бюджетное профессиональное образовательное учреждение

«Нижегородский Губернский колледж»

**ОТЧЕТ**

**ПО ВЫПОЛНЕНИЮ ЗАДАНИЯ ПРОИЗВОДСТВЕННОЙ ПРАКТИКИ**

ПМ.02 ОСУЩЕСТВЛЕНИЕ ИНТЕГРАЦИИ ПРОГРАММНЫХ МОДУЛЕЙ

|  |  |
| --- | --- |
| Руководители\_\_\_\_\_\_\_Мамшева Ю.С., Голубева Е.П. | 06.06.2025 |
| Студент\_\_\_\_\_\_\_\_\_\_\_\_\_\_Руфкин М.Н. | 06.06.2025 |
| Специальность, группа: 09.02.07, 33П |  |

Нижний Новгород

2025

**СОДЕРЖАНИЕ**

[**ВВЕДЕНИЕ** 3](#_Toc200540747)

[**ТЕОРЕТИЧЕСКАЯ ЧАСТЬ** 5](#_Toc200540748)

[**1.1 Организационная структура предприятия** 5](#_Toc200540749)

[**1.2 Анализ программного обеспечения и технического оснащения** 6](#_Toc200540750)

[**1.3 Описание инструментальных средств разработки и сопровождения программных средств** 8](#_Toc200540751)

[**ПРАКТИЧЕСКАЯ ЧАСТЬ** 9](#_Toc200540752)

[**2.1 Анализ предметной области** 9](#_Toc200540753)

[**2.2 Проектирование программного продукта** 15](#_Toc200540754)

[**2.3 Разработка программного модуля** 23](#_Toc200540755)

[**2.4 Тестирование программного продукта** 24](#_Toc200540756)

[**ЗАКЛЮЧЕНИЕ** 25](#_Toc200540757)

[**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ** 26](#_Toc200540758)

[**ПРИЛОЖЕНИЯ** 27](#_Toc200540759)

[**ПРИЛОЖЕНИЕ А** 27](#_Toc200540760)

[**ПРИЛОЖЕНИЕ Б** 28](#_Toc200540761)

[**ПРИЛОЖЕНИЕ В** 40](#_Toc200540762)

# **ВВЕДЕНИЕ**

В условиях цифровой трансформации страхового рынка, а также расширении компании, количество необходимых сотрудников для выполнения обязанностей возрастает. И одним из актуальных и удобных методов получения квалифицированного сотрудника является стажерская программа. В связи с этим можно выделить несколько основных моментов для компании СПАО «Ингосстрах»:

* Рост численности стажёров.
* Неэффективность текущих методов обучения (разрозненные Excel-отчёты, отсутствие единой базы знаний).
* Требования регуляторов.

Целью данного проекта является разработка и внедрение системы Intern Guide – цифровой платформы для адаптации, обучения и мониторинга стажёров компании СПАО «Ингосстрах». Система преследует три ключевые цели:

1. Оптимизация адаптации сокращение периода ввода в должность с 3 до 2 месяцев
2. Стандартизация обучения создание единой базы знаний по:

* Страховым продуктам (ОСАГО, КАСКО, ДМС)
* Корпоративным стандартам
* CRM-системам (1С:Страхование)

1. Автоматизация контроля переход от Excel-отчетов к цифровому мониторингу:

* Прогресса обучения
* Результатов тестирования
* Обратной связи от кураторов

**Объект исследования**: Процесс адаптации стажёров в ПАО «Ингосстрах».

**Предмет исследования**: Автоматизированная система управления обучением (LMS).

Для реализации программного продукта и выполнения поставленной задачи по разработке информационной системы, необходимо следовать следующим этапам решения:

1. Анализ требований:

* Опрос сотрудников компании

1. Проектирование:

* Прототипирование UI в Figma
* ER-диаграммы в Drowio
* UML-диаграммы

1. Реализация:

* Разработка на Python, JS, HTML.

# **ТЕОРЕТИЧЕСКАЯ ЧАСТЬ**

## **1.1 Организационная структура предприятия**

«Ингосстрах» – крупнейшая страховая компания в России, основанная в 1947 году. Основная задача данной компании – обеспечить своим клиентам финансовую защиту и душевное спокойствие.

Организационная структура предприятия строится следующим образом (Приложение А):

* Центральный офис: Москва, отвечает за общее управление дочерними предприятиями и офисами, а также занимается принятием стратегических решений в области осуществления каких-либо действий во внутреннем строе компании и внешних взаимодействий с другими компаниями.
* Региональные дирекции: в крупных городах находятся 8 региональных представительств, которые отвечают за развитие бизнеса и своевременное обслуживание клиентов, и принятие необходимых решений во всех регионах России.
* Филиалы и представительства: компания имеет более 80 различных филиалов и представительств по всей стране.

Компания «Ингосстрах» осуществляет свою деятельность в различных направлениях. Среди них можно выделить следующие основные бизнес-процессы, которые необходимы для выполнения грамотного обслуживания клиентов и полного удовлетворения их потребностям:

* Андеррайтинг: определение и оценка потенциальных и фактических рисков, на основе которых определяются условия страхования.
* Заключения договоров: оформление страховых полисов на различные виды имущества и личное здоровье.
* Урегулирование убытков: анализ и обработка страховых случаев, и выплата материальных компенсаций.
* Инвестиционная деятельность: управление страховыми резервами и инвестирование средств.
* Клиентское обслуживание: обслуживание клиентов, консультирование по вопросам касающихся области деятельности предприятия, а также решение их проблем.

«Ингосстрах» – крупная и динамично развивающаяся страховая компания с многоуровневой организационной структурой и сложными бизнес-процессами. Для эффективного функционирования компании необходимы надежные и эффективные информационные системы, в том числе базы данных.

## **1.2 Анализ программного обеспечения и технического оснащения**

В современном мире невозможно представить работу какой-либо компании без использования программного обеспечения. Оно нужно для удобного и автоматизированного использования информации и ресурсов. Компания «Ингосстрах», занимающаяся страховой деятельностью, также использует большое количество разнообразного ПО. Вот примеры некоторых из тех, что используются сотрудниками:

* Автоматизированная информационная система (АИС) — это совокупность устройств и программных средств, объединённых информационными потоками и предназначенных для сбора, хранения, обработки и выдачи информации с целью автоматизации различных процессов. Данная программа помогает управлять ресурсами компании, анализировать данные о продажах и клиентах, планировать бюджет и оптимизировать бизнес-процессы. Также она используется для обработки платежей, управления счетами, выдачи кредитов, оценки рисков и многого другого.
* Microsoft Outlook — компьютерная программа-органайзер с функциями почтового клиента компании «Майкрософт», входящая в пакет офисных программ Microsoft Office.  Используется для работы с электронными письмами и составления списков задач и конктактов.

Если рассматривать используемое ПО с точки зрения хранения и использования данных, то используемое программное обеспечение представляет собой следующий список:

1. **Программное обеспечение**

Операционная система: Windows

СУБД: Oracle Database (Oracle Database Enterprise Edition)

Средства разработки баз данных: Oracle SQL Developer

Инструменты администрирования баз данных: Oracle Enterprise Manager

Системы резервного копирования и восстановления данных: Oracle Recovery Manager (RMAN), Veritas NetBackup

Системы управления полисами и договорами: собственная разработка на базе Oracle Database

Система урегулирования убытков: собственная разработка на базе Oracle Database

Система управления рисками: собственная разработка на базе Oracle Database

Система финансового учета: SAP ERP

Система клиентского обслуживания (CRM): Salesforce

1. **Техническое оснащение**

Серверы: Dell PowerEdge, HPE ProLiant

Хранилища данных: Dell EMC VMAX, HPE 3PAR

Сетевое оборудование: Cisco, Juniper Networks

Системы бесперебойного питания: APC, Eaton

Системы кондиционирования и охлаждения: Liebert, Stulz

Компания «Ингосстрах» использует надежное и современное программное обеспечение, и техническое оснащение для обеспечения стабильной и эффективной работы своих информационных систем.

## **1.3 Описание инструментальных средств разработки и сопровождения программных средств**

Для создания системы Intern Guide выбран современный технологический стек, ориентированный на высокую производительность, масштабируемость и интеграцию с существующей ИТ-инфраструктурой СПАО «Ингосстрах».

1. **Backend**

Backend-разработка реализуется на Python с использованием следующих ключевых компонентов:

* Язык программирования: Python 3.11
* Фреймворк: FastAPI – обеспечивает высокую скорость обработки запросов и автоматическую генерацию OpenAPI-документации
* База данных: PostgreSQL 15 с расширением TimescaleDB для хранения временных рядов

1. **Frontend**

Frontend-разработка строится на классическом веб-стеке:

* JavaScript: ES6+ с использованием модульного подхода
* HTML5: Семантическая верстка
* CSS3: методология BEM для стилей
* Фреймворк: React 18

Для обеспечения качества используются несколько уровней тестирования:

1. Модульные тесты:

* Python: pytest с покрытием >85%

1. Интеграционные тесты:

* Тестирование API через Postman (коллекции в Git)

1. Нагрузочное тестирование:

* Проверка отказоустойчивости (chaos engineering)

# **ПРАКТИЧЕСКАЯ ЧАСТЬ**

## **2.1 Анализ предметной области**

В современных корпоративных условиях эффективная адаптация новых сотрудников играет ключевую роль в их дальнейшей продуктивности. Система Intern Guide призвана структурировать процесс обучения, предоставляя стажёрам доступ к учебным материалам, тестам и аналитике их прогресса.

Система Intern Guide должна предоставлять стажёрам доступ к учебным материалам, включая статьи, гайды и инструкции. Эти материалы необходимо систематизировать по темам и уровням сложности, чтобы облегчить процесс обучения. Важной функцией является возможность загрузки различных форматов файлов, таких как PDF, презентации и видео, что сделает обучение более наглядным и удобным.

Тестирование и проверка знаний — ещё один ключевой аспект системы. Система должна автоматически проверять результаты и формировать отчёты, что позволит оперативно оценивать успеваемость стажёров. Для мониторинга прогресса необходимо предусмотреть трекер активности, который будет фиксировать пройденные материалы и результаты тестов.

Обратная связь и коммуникация также играют важную роль в системе. Стажёры должны иметь возможность комментировать материалы и задавать вопросы, а кураторы — оценивать выполненные задания и давать рекомендации. Для удобства обсуждений можно предусмотреть чат или форум, где участники смогут обмениваться опытом и советами.

Главными пользователями будут:

1. **Стажёры** – основная целевая аудитория, которым необходим доступ к учебным материалам, возможности проходит тестирование и отслеживать личный прогресс.
2. **Кураторы** – эксперты, которые помогают стажерам проходить процесс обучения. Они должны публиковать учебные материалы, создавать тесты, отвечать на вопросы.
3. **Администраторы** – управляют пользователями и контентом.

Система Intern Guide должна предоставлять следующий функционал:

1. **Управление учебными материалами**

* Публикация статей, гайдов, инструкций.
* Категоризация контента (по темам, уровням сложности).
* Возможность загрузки файлов (PDF, презентации, видео).

1. **Тестирование и проверка знаний**

* Создание тестов с различными типами вопросов (множественный выбор, открытые вопросы).
* Автоматическая проверка результатов.
* Формирование отчётов по успеваемости.

1. **Мониторинг прогресса стажёров**

* Трекер активности (пройденные материалы, результаты тестов).
* Визуализация статистики (графики, диаграммы).
* Система рейтингов и достижений.

1. **Обратная связь и коммуникация**

* Комментарии к материалам.
* Оценка заданий кураторами.
* Чат или форум для обсуждений.

После того, как это всё было определено, можно приступить ко второму этапу, который включает в себя выявление информационных объектов (сущностей) и связей между ними. Также для каждого объекта определяется набор характеристик (данных, которые находятся в каждом объекте).

В результате проведенного анализа потребностей и требований к системе были выявлены основные сущности. Все сущности базы данных представлены в виде таблиц, хранящих определенные данные.

Основными сущностями системы являются пользователи, учебные материалы, тесты, вопросы и данные о прогрессе. Пользователи делятся на стажёров, кураторов и администраторов. Учебные материалы включают статьи, гайды и другие ресурсы. Тесты и вопросы позволяют проверять уровень усвоения материала, а данные о прогрессе фиксируют результаты. Основные характеристики каждой сущности описаны в таблицах ниже.

**Таблица 1. Атрибуты таблицы "Пользователь" (User)**

| **Тип поля** | **Название** | **Назначение** |
| --- | --- | --- |
| ПК | user\_id | Уникальный идентификатор пользователя (целое число) |
|  | name | Имя пользователя (строка) |
|  | surname | Фамилия пользователя (строка) |
|  | login | Уникальный логин для входа в систему (строка) |
|  | password | Хешированный пароль для авторизации (строка) |
|  | role | Роль пользователя в системе (строка) |
|  | registration\_date | Дата регистрации пользователя (дата) |

**Таблица 2. Атрибуты таблицы "Учебные материалы" (Materials)**

| **Тип поля** | **Название** | **Назначение** |
| --- | --- | --- |
| ПК | material\_id | Уникальный идентификатор материала (целое число) |
|  | title | Наименование материала (строка) |
| ВК | author\_id | Ссылка на вид автора (целое число) |
|  | content | Текст статьи или описание (текст) |
|  | category | Категория (строка) |
|  | upload\_date | Дата публикации (дата) |
|  | file\_url | Ссылка на файлы при необходимости (строка) |

**Таблица 3. Атрибуты таблицы "Тесты" (Tests)**

| **Тип поля** | **Название** | **Назначение** |
| --- | --- | --- |
| ПК | test\_id | Уникальный идентификатор теста (целое число) |
|  | title | Заголовок (строка) |
| ВК | creator\_id | Автор теста (целое число) |
|  | description | Описание (строка) |
|  | max\_score | Максимальный балл за тест (целое число) |

**Таблица 4. Атрибуты таблицы "Вопросы" (Questions)**

| **Тип поля** | **Название** | **Назначение** |
| --- | --- | --- |
| ПК | question\_id | Уникальный идентификатор вопроса (целое число) |
| ВК | test\_id | Ссылка на тест (целое число) |
|  | text | Содержание вопроса (строка) |
|  | points | Баллы за вопрос (целое число) |
|  | result | Правильный ответ (строка) |

**Таблица 5. Атрибуты таблицы "Прогресс" (Progress)**

| **Тип поля** | **Название** | **Назначение** |
| --- | --- | --- |
| ПК | progress\_id | Уникальный идентификатор прогресса (целое число) |
| ВК | user\_id | Ссылка на пользователя (целое число) |
| ВК | test\_id | Ссылка на тест (целое число) |
|  | completion\_date | Дата завершения (дата) |
|  | score | Набранные баллы (целое число) |
|  | status | Статус (строка) |

Связи между сущностями обеспечивают целостность системы. Например, один пользователь может создавать несколько учебных материалов или тестов, а каждый тест включает множество вопросов. Данные о прогрессе связывают пользователей с пройденными тестами и материалами, что позволяет отслеживать индивидуальные достижения. Список связей между информационными объектами данной системы:

1. Пользователь – Материал: один ко многим (один пользователь может создать несколько материалов).
2. Пользователь – Тест: один ко многим (куратор создаёт тесты).
3. Тест – Вопрос: один ко многим (в одном тесте несколько вопросов).
4. Пользователь – Прогресс: один ко многим (у одного пользователя множество записей о прогрессе).

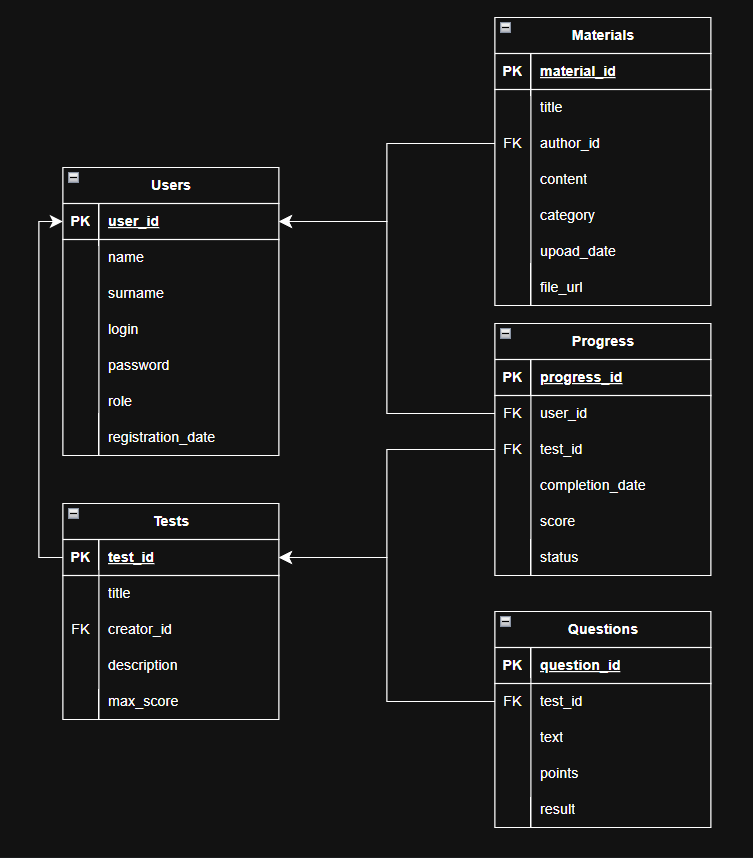
И в заключении приступаем к третьему этапу анализа предметной области – проектирование и разработка информационной модели предметной области, которая используется для структурирования данных внутри БД. Самой популярной моделью является ER-модель.

Особенности проектируемой модели:

* Простота и интуитивная понятность
* Минимальное количество сущностей
* Четкие взаимосвязи между объектами
* Возможность последующего масштабирования
* Оптимизация для мобильного использования

ER-диаграмма будет включать все перечисленные сущности с указанием типов связей между ними и ключевых атрибутов каждой сущности.

На рисунке ниже представлена диаграмма для текущей системы.

****

**Рисунок 1. ER-диаграмма для текущей системы**

## **2.2 Проектирование программного продукта**

UML (Unified Modeling Language) — это стандартизированный язык графического моделирования, используемый в объектно-ориентированном анализе и проектировании. UML-диаграммы помогают визуализировать, проектировать и документировать структуру и поведение программных систем.

Существует 14 типов UML-диаграмм, которые делятся на две группы:

1. **Структурные диаграммы** (отображают статическую структуру системы).
2. **Поведенческие диаграммы** (описывают динамику работы системы).

Самыми основными из них выступают 8 диаграмм: классов, кооперации, деятельности, компонентов, последовательности, потоков данных, развертывания, состояния, вариантов использования.

1. **Диаграмма классов**

Тип: структурная

Назначение: отображает статическую структуру системы в виде классов, их атрибутов, методов и взаимосвязей.

Основные элементы:

* Класс: Прямоугольник с тремя секциями:
* Имя класса
* Атрибуты
* Методы
* Связи: наследование, ассоциации, агрегация/композиция.

1. **Диаграмма последовательности**

Тип: Поведенческая

Назначение: показывает взаимодействие объектов в хронологическом порядке (как сообщения передаются между ними).

Основные элементы:

* Объекты – участники взаимодействия
* Линии жизни – вертикальные пунктирные линии, показывающие время жизни объекта
* Сообщения – стрелки между линиями жизни
* Активационные полосы – прямоугольники на линиях жизни, показывающие активность объекта

1. **Диаграмма развертывания**

Тип: Структурная

Назначение: показывает физическое размещение компонентов системы на серверах, узлах сети и устройствах.

Основные элементы:

* Узлы – физические устройства
* Артефакты – развертываемые компоненты
* Связи – показывают коммуникацию между узлами

1. **Диаграмма состояний**

Тип: Поведенческая

Назначение: описывает жизненный цикл объекта, показывая, как он переходит между состояниями под воздействием событий.

Основные элементы:

* Состояние – прямоугольник с закругленными углами
* Переход – стрелка с событием
* Начальное и конечное состояния – черные круги

1. **Диаграмма вариантов использования**

Тип: Поведенческая

Назначение: описывает взаимодействие между пользователями (акторами) и системой через сценарии использования.

Основные элементы:

* Актор – роль, взаимодействующая с системой
* Вариант использования – овал с названием действия
* Связи

1. **Диаграмма кооперации**

Тип: Поведенческая

Назначение: показывает взаимодействие объектов для выполнения конкретного сценария через обмен сообщениями.

Основные элементы:

* Объекты – прямоугольники с подчеркнутыми именами
* Сообщения – стрелки с номерами

1. **Диаграмма деятельности**

Тип: Поведенческая

Назначение: описывает бизнес-процессы и алгоритмы в виде последовательности действий.

Основные элементы:

* Действия – скругленные прямоугольники
* Ветвления – ромбы
* Потоки управления – стрелки

1. **Диаграмма компонентов**

Тип: Структурная

Назначение: отображает физические компоненты системы (модули, библиотеки) и их зависимости.

Основные элементы:

* Компоненты – прямоугольники с иконкой
* Интерфейсы – «лоллипопы»
* Зависимости – пунктирные стрелки

1. **Диаграмма потоков данных**

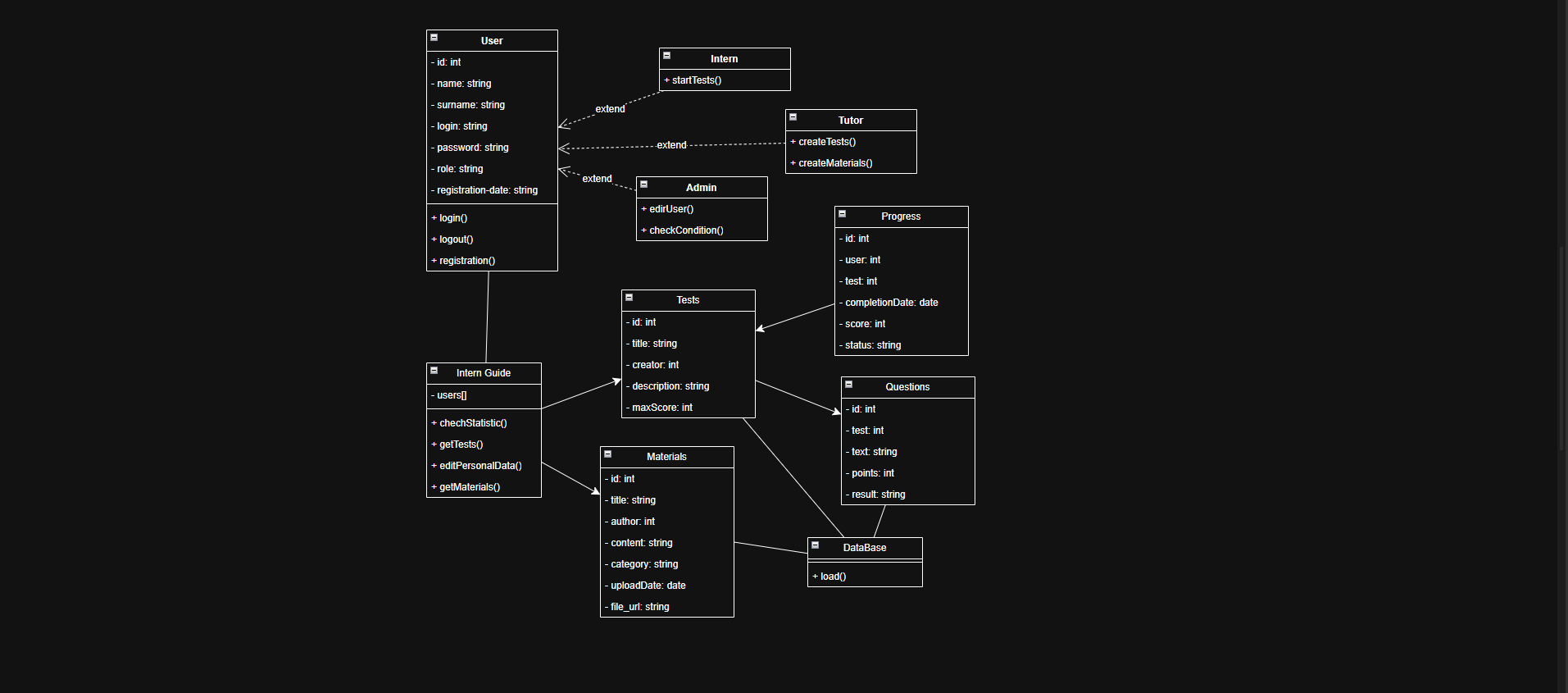
Тип: Структурная

Назначение: показывает движение информации между процессами, хранилищами и внешними сущностями.

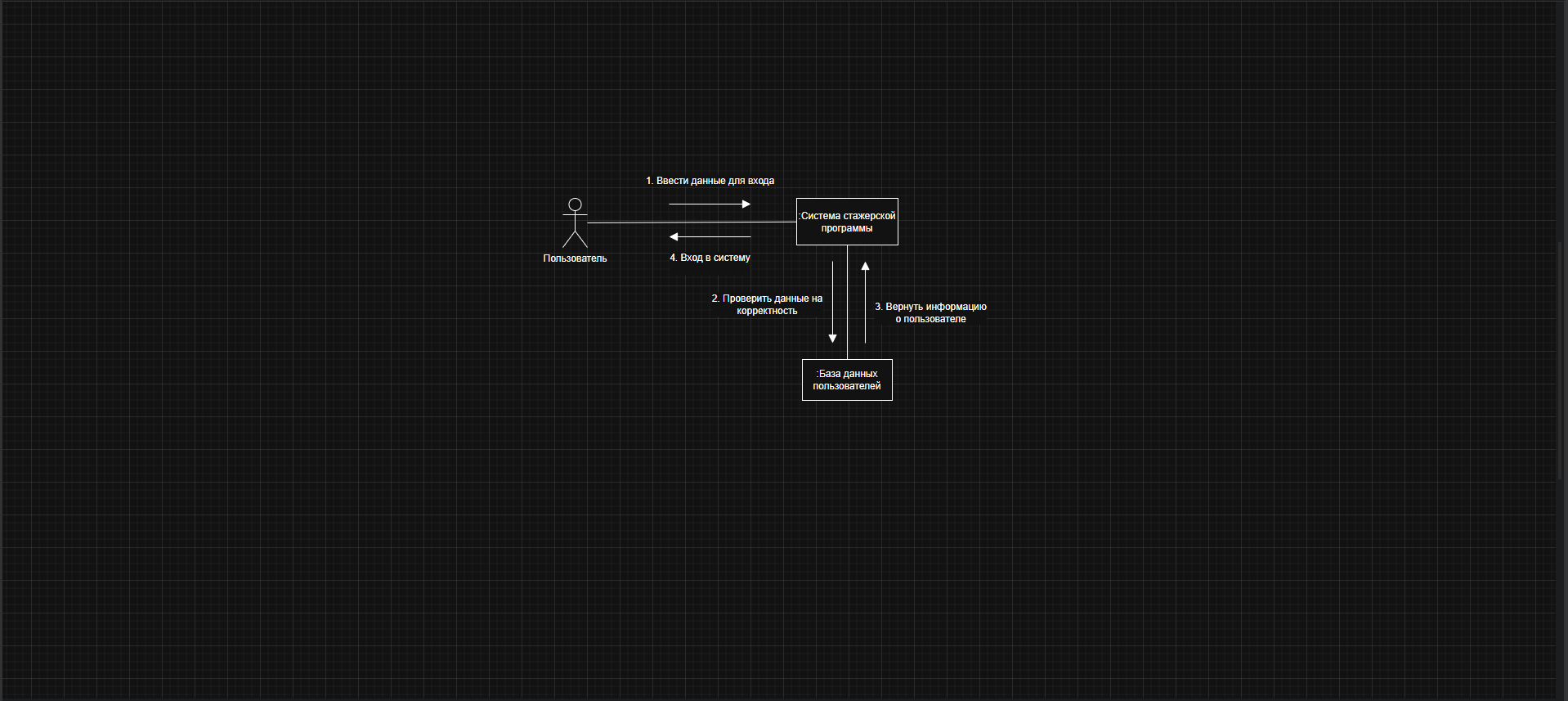
Основные элементы:

* Процессы – круги
* Хранилища – две параллельные линии
* Потоки данных – стрелки с подписями

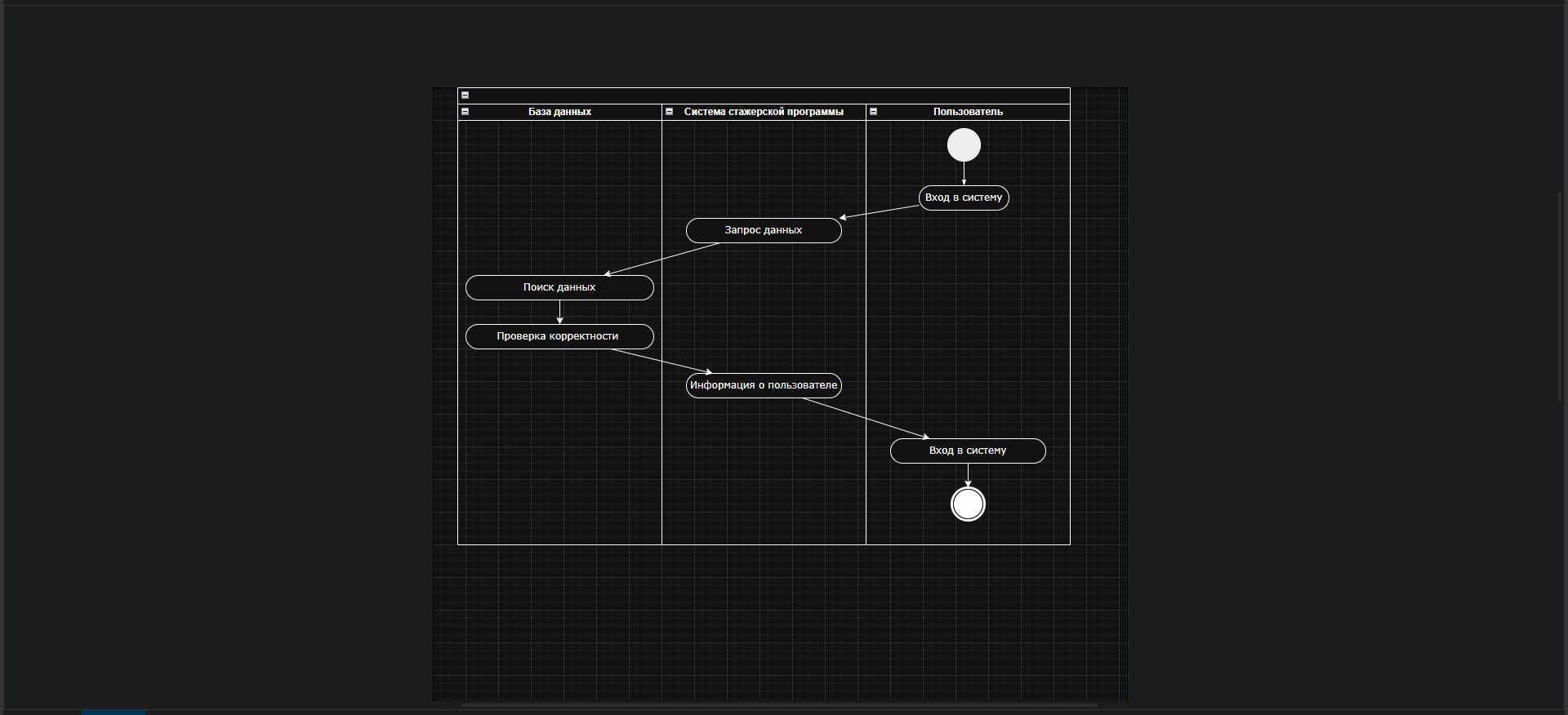
Для описания поведения и работы информационной системы Intern Guide, которая помогает стажерам лучше осваиваться на новом месте и получать различные советы, были разработаны следующие несколько диаграмм:

****

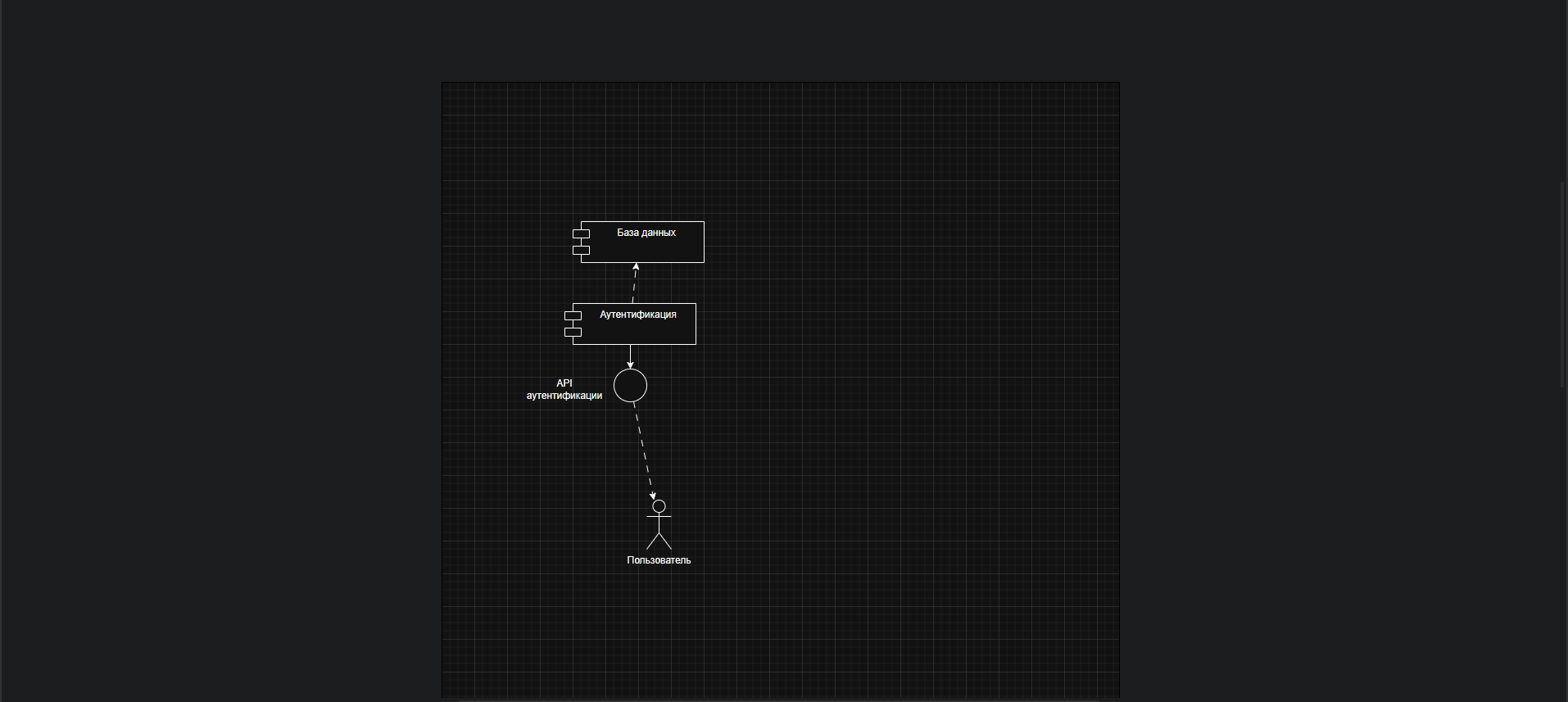
**Рисунок 2. Авторизация - диаграмма классов**

****

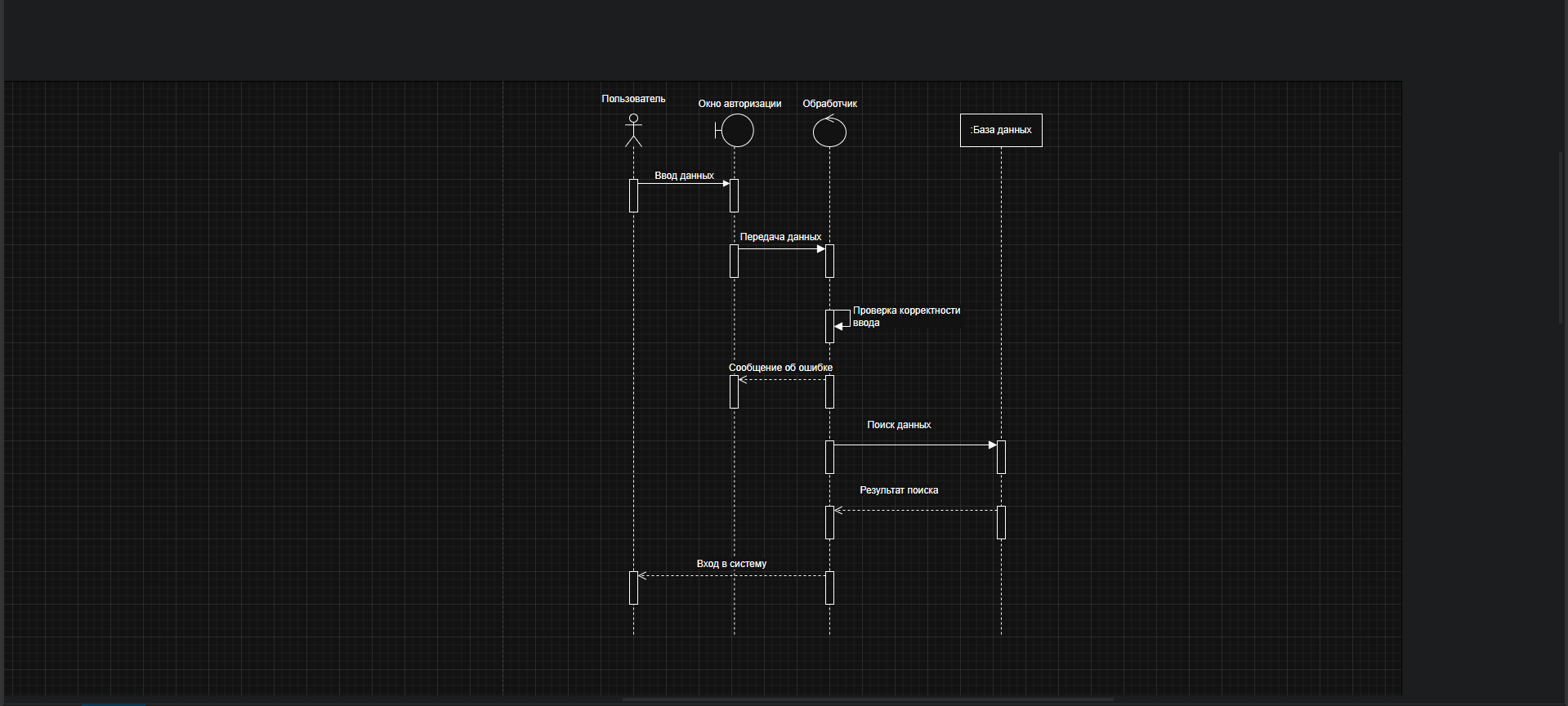
**Рисунок 3. Авторизация - диаграмма коопераций**

****

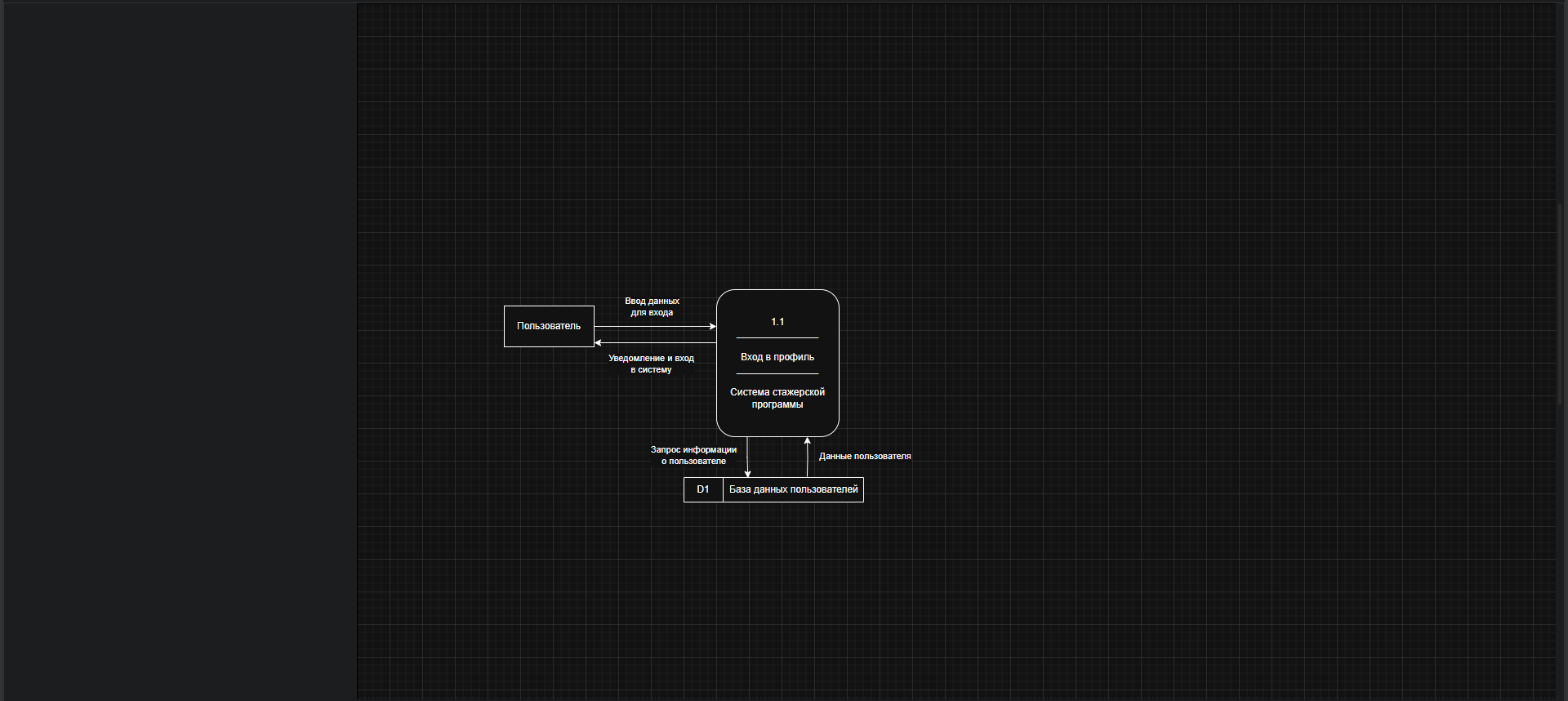
**Рисунок 4. Авторизация - диаграмма деятельности**

****

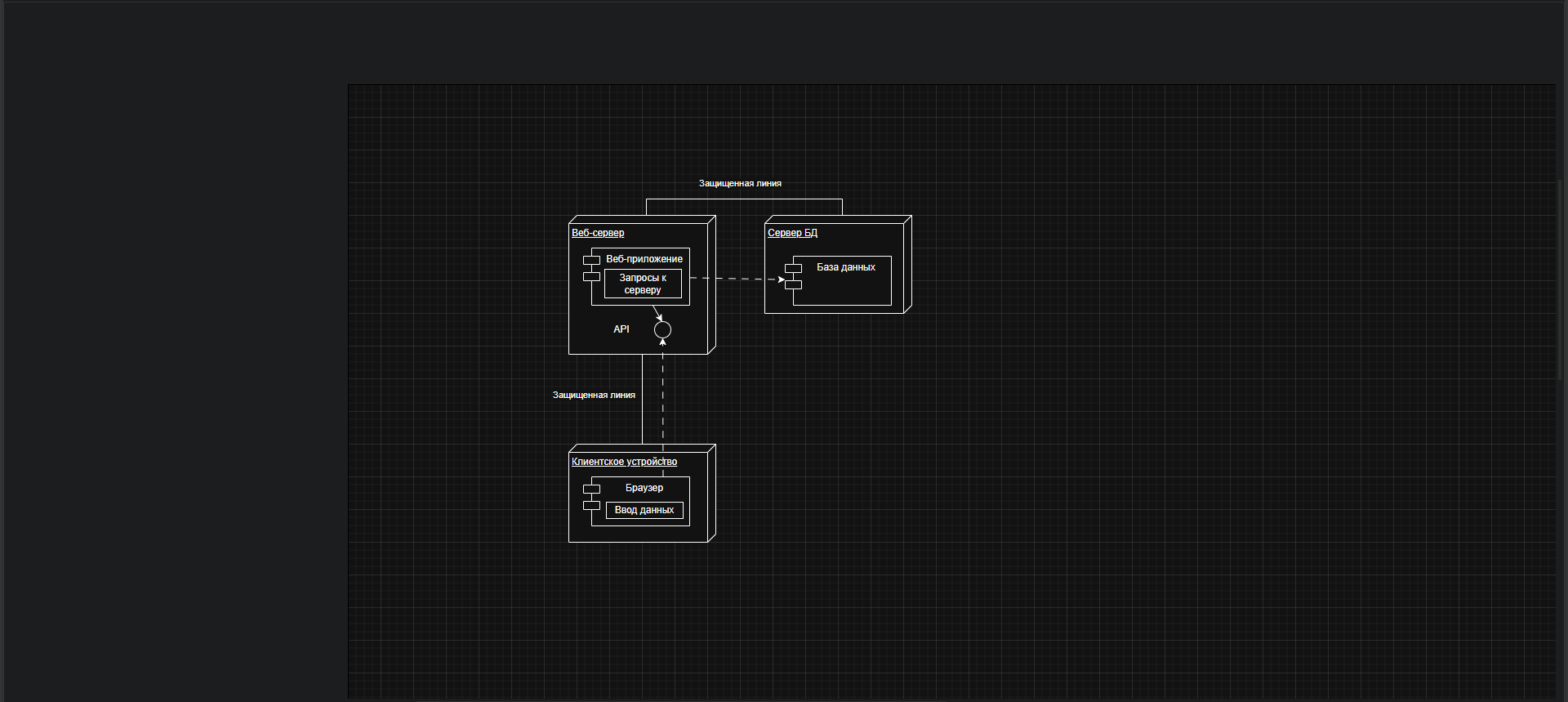
**Рисунок 5. Авторизация - диаграмма компонентов**

****

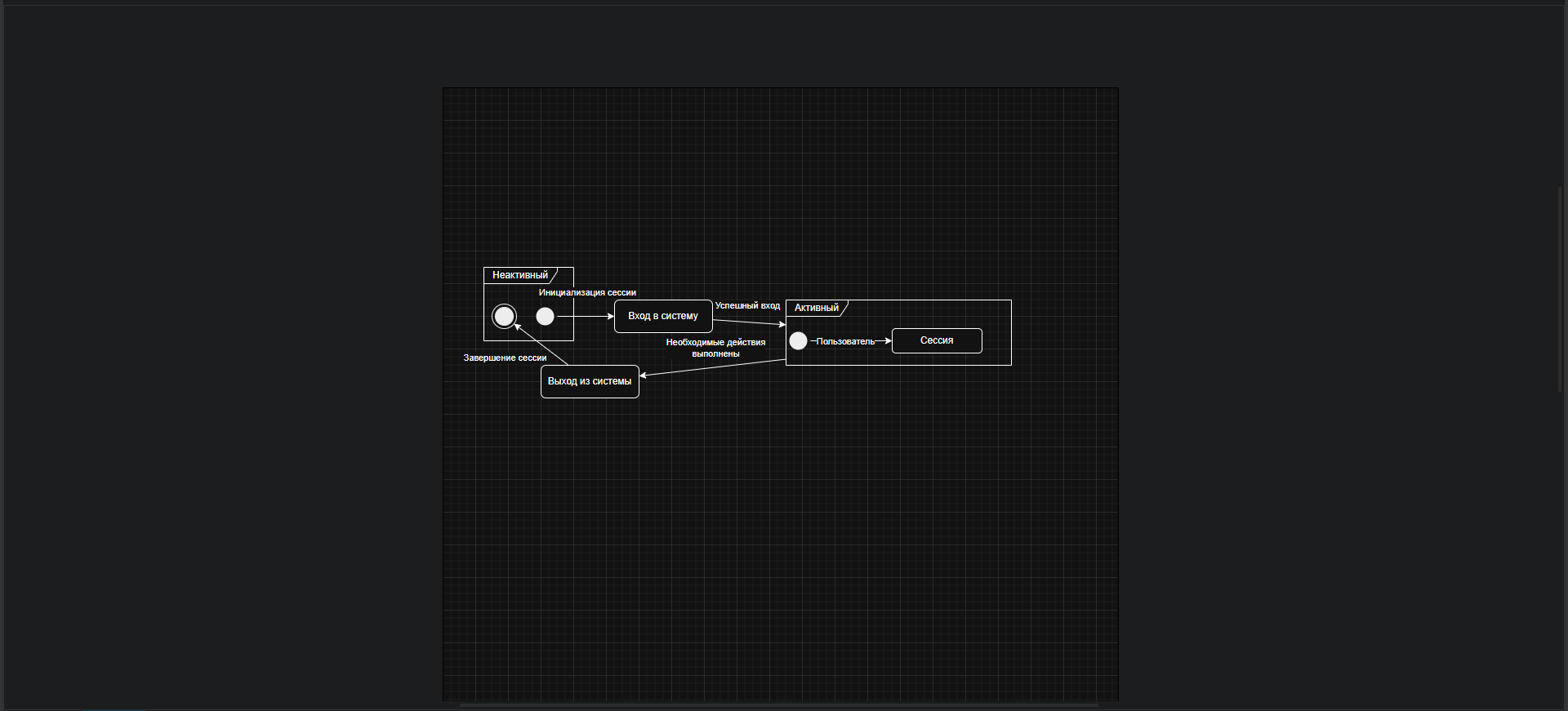
**Рисунок 6. Авторизация - диаграмма последовательности**

****

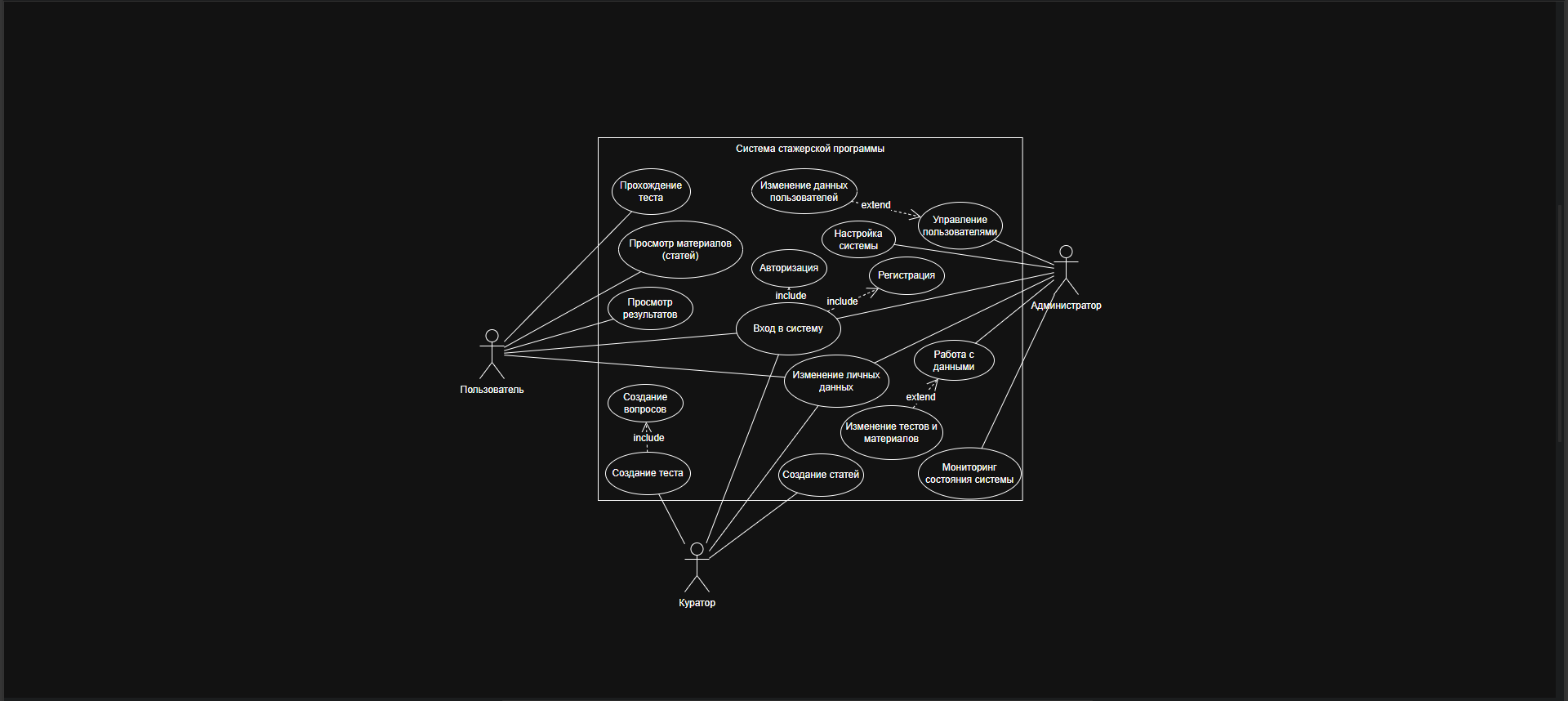
**Рисунок 7. Авторизация - диаграмма потоков данных**

****

**Рисунок 8. Авторизация - диаграмма развертывания**

****

**Рисунок 9. Авторизация - диаграмма состояния**

****

**Рисунок 10. Авторизация - диаграмма вариантов использования**

Дополнительные диаграммы для остальных модулей разрабатываемой системы находятся в Приложении Б.

Данный набор диаграмм обеспечивает полное визуальное представление системы на всех уровнях — от логики работы до физической инфраструктуры.

## **2.3 Разработка программного модуля**

Тестовая документация – это критически важный компонент процесса

## **2.4 Тестирование программного продукта**

PlantDoctor — это веб-приложение для ухода за растениями,

# **ЗАКЛЮЧЕНИЕ**

Работа над проектом Intern Guide для СПАО «Ингосстрах» стала важным этапом в профессиональном развитии как будущего специалиста в области разработки программного обеспечения. В процессе проектирования и реализации системы были углублены знания в современных технологиях веб-разработки, получив ценный практический опыт.

На этапе проектирования особое внимание было уделено анализу требований заказчика. Это позволило создать систему, которая не просто соответствует формальным требованиям, но и реально решает проблемы пользователей. Разработка UML-диаграмм помогла четко структурировать архитектуру приложения до начала программирования.

Практическая реализация проекта велась с использованием современного технологического стека. Для backend-части был выбран Python с фреймворком FastAPI, что обеспечило высокую производительность API и простоту интеграции с существующей инфраструктурой компании. Frontend-часть разрабатывалась на React с TypeScript, что позволило создать интуитивно понятный интерфейс с поддержкой всех необходимых функций.

Важным аспектом проекта стало тестирование системы. Реализация модульных, интеграционных и нагрузочных тестов позволила определить все ошибки и недочеты на всех этапах создания системы.

В результате была создана система, которая успешно решает поставленные задачи: сокращает время адаптации новых сотрудников, автоматизирует процессы обучения и предоставляет аналитические инструменты для кураторов.

Однако также выявились направления для дальнейшего профессионального роста, такие как углубленное изучение систем очередей сообщений (Kafka), освоение Infrastructure as Code (Terraform) и применение методов машинного обучения для анализа данных.

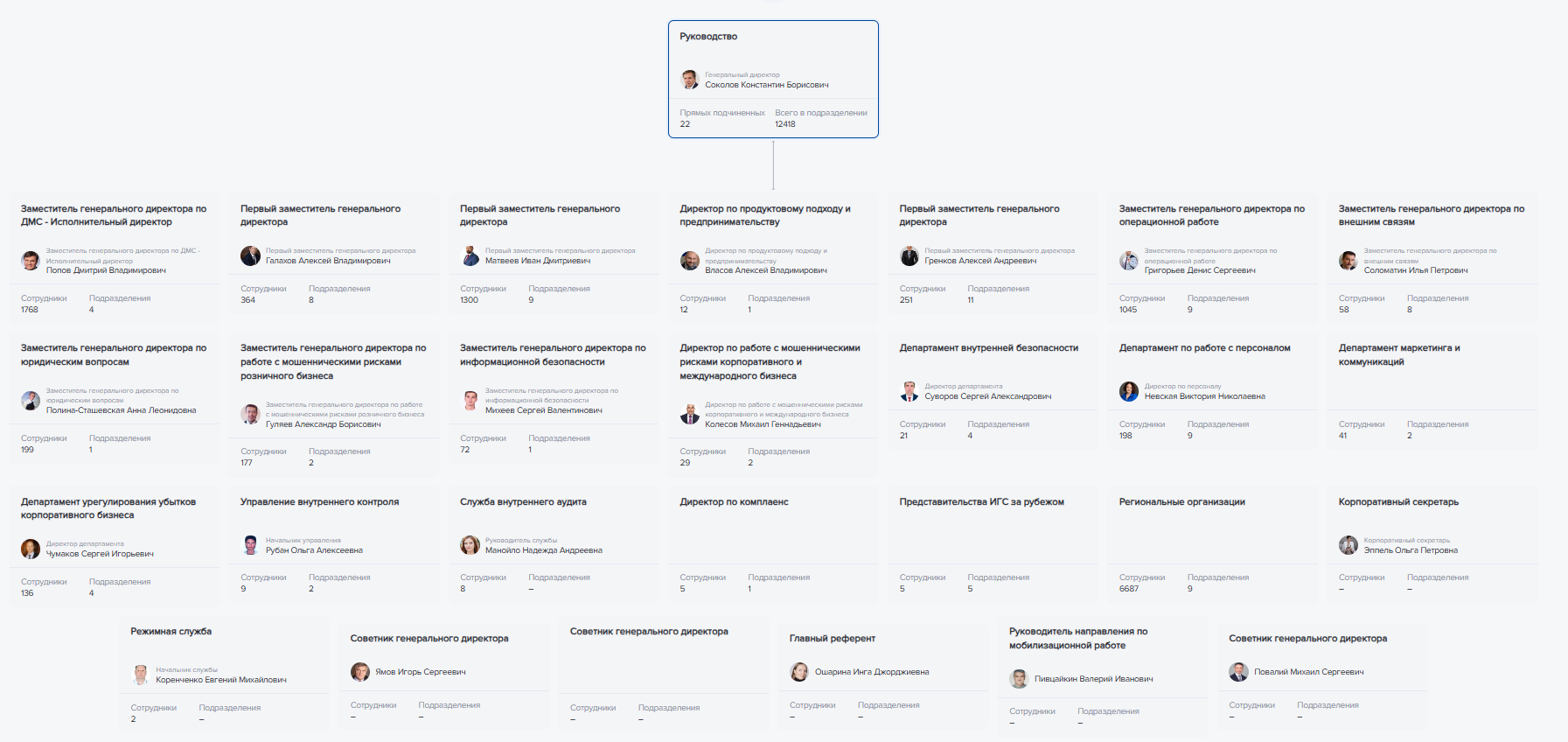
# **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Современный UX/UI-дизайн / Аарон Уолтер. — М.: Манн, 2021.
2. Тестирование ПО / Роман Савин. — СПб.: БХВ, 2019.
3. UML 2.5. Подробное руководство / Лешек Мацяшек. — ДМК, 2022.
4. М. Гринвальд, С. Харрисон. Flask. Разработка веб-приложений на Python. — СПб.: Питер, 2021. — 352 с.
5. Д. Грубер. SQLAlchemy. Профессиональное использование. — М.: ДМК Пресс, 2020. — 488 с.
6. А. Швец. Python и тестирование. Полное руководство. — 2-е изд. — М.: БХВ-Петербург, 2022. — 672 с.
7. SQL для анализа данных / Кэтрин Тан. — O’Reilly, 2021.
8. Системное мышление / Доннелла Медоуз. — Манн, 2020.
9. Р. Савин. Тестирование DOT COM. — 5-е изд. — М.: Альфа-книга, 2023. — 816 с.
10. Д. Флэнаган. JavaScript. Подробное руководство. — 7-е изд. — СПб.: Символ-Плюс, 2022. — 1088 с.
11. А. Купер. Интерфейс. Основы проектирования взаимодействия. — 4-е изд. — СПб.: Питер, 2020. — 496 с.
12. Р. Мартин. Чистый код. Создание, анализ и рефакторинг. — М.: Вильямс, 2022. — 464 с.
13. Дж. Уиттакер. Как тестируют в Google. — М.: Альпина Паблишер, 2021. — 382 с.
14. В. Дронов. HTML5, CSS3 и JavaScript. Исчерпывающее руководство. — 5-е изд. — СПб.: БХВ-Петербург, 2023. — 1088 с.
15. М. Фаулер. Рефакторинг. Улучшение существующего кода. — 2-е изд. — М.: Вильямс, 2020. — 448 с.
16. К. Хорстманн. Java Script для профессионалов. — М.: Диалектика, 2022. — 1088 с.

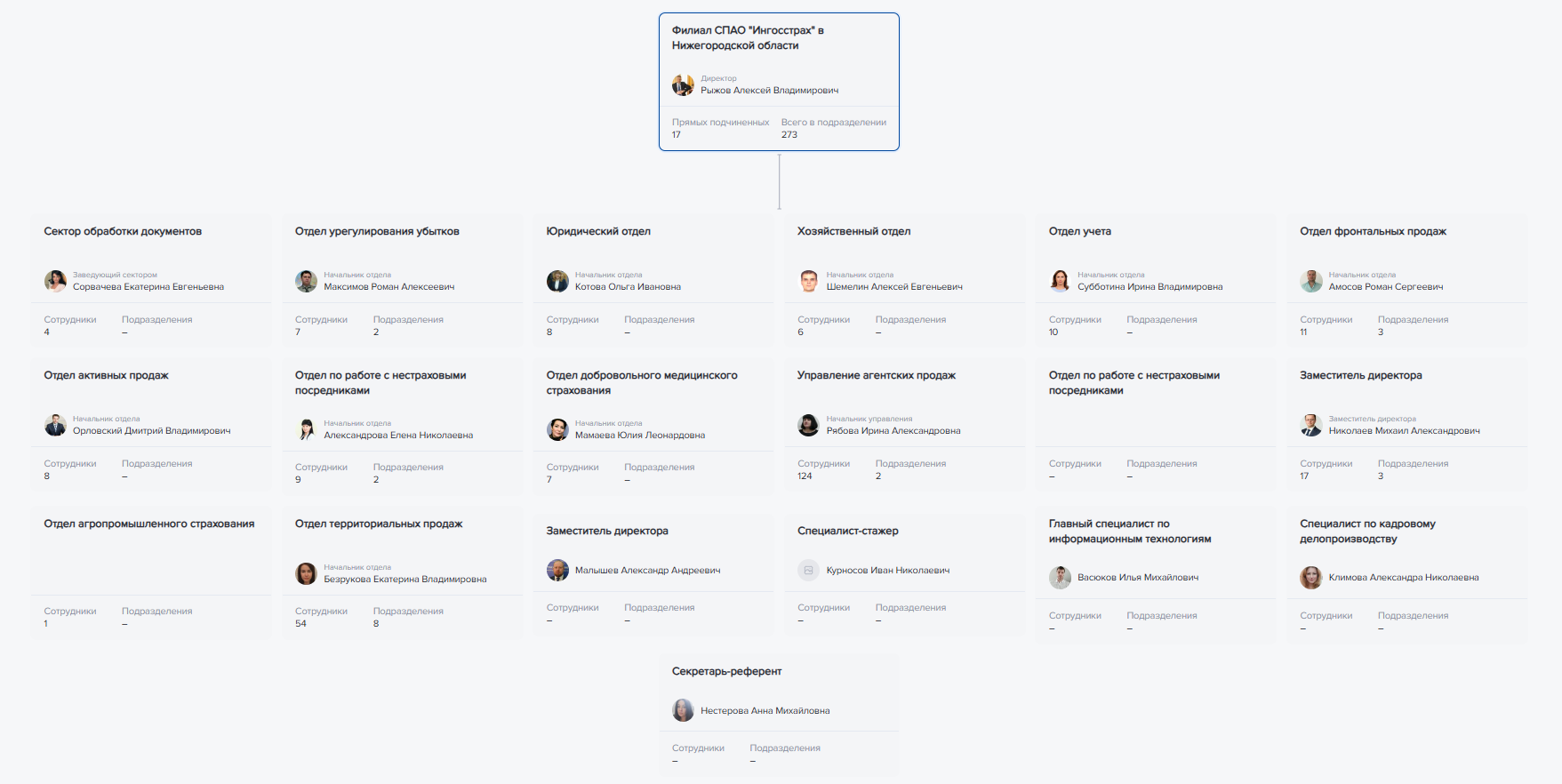
# **ПРИЛОЖЕНИЯ**

## **ПРИЛОЖЕНИЕ А**

Оргструктура предприятия

****

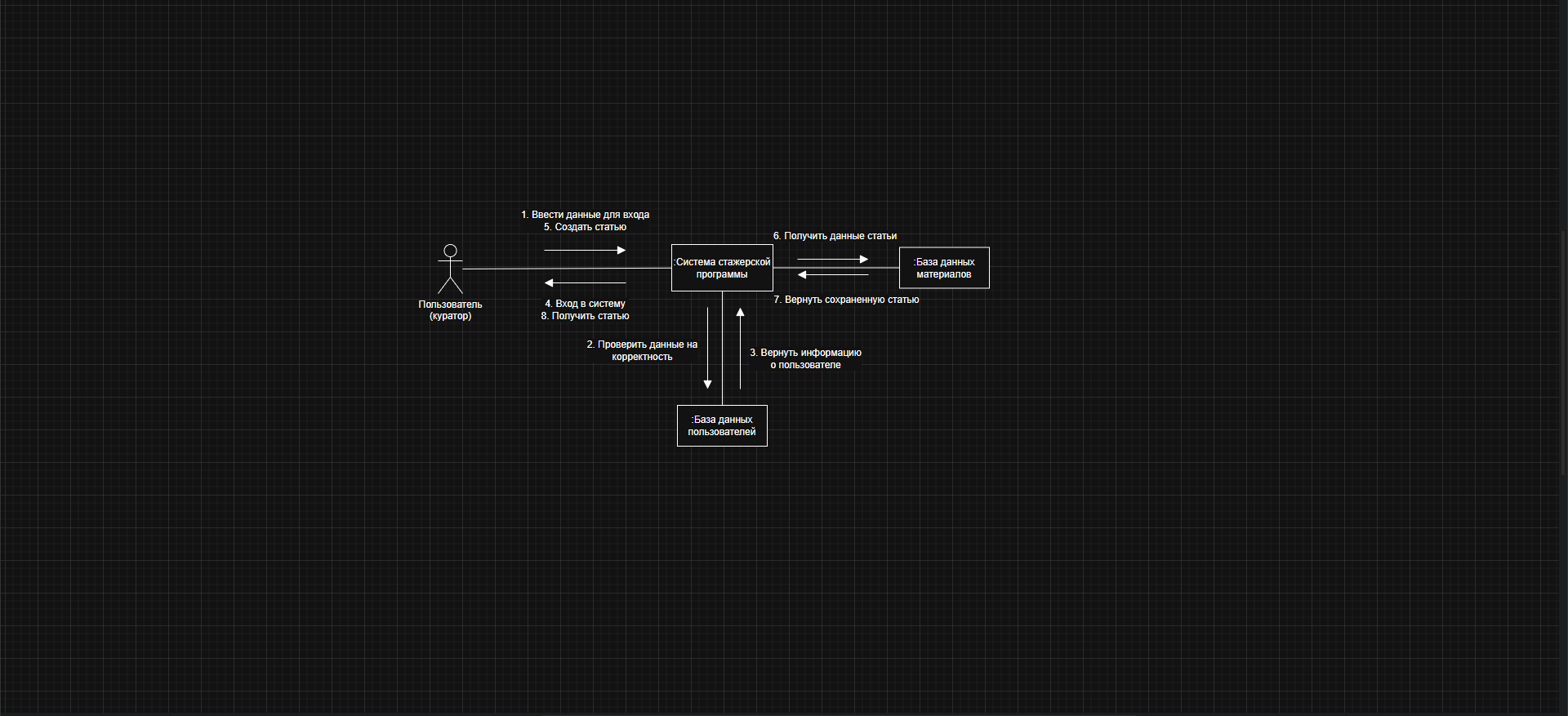
**Рисунок 11. Структура предприятия**

****

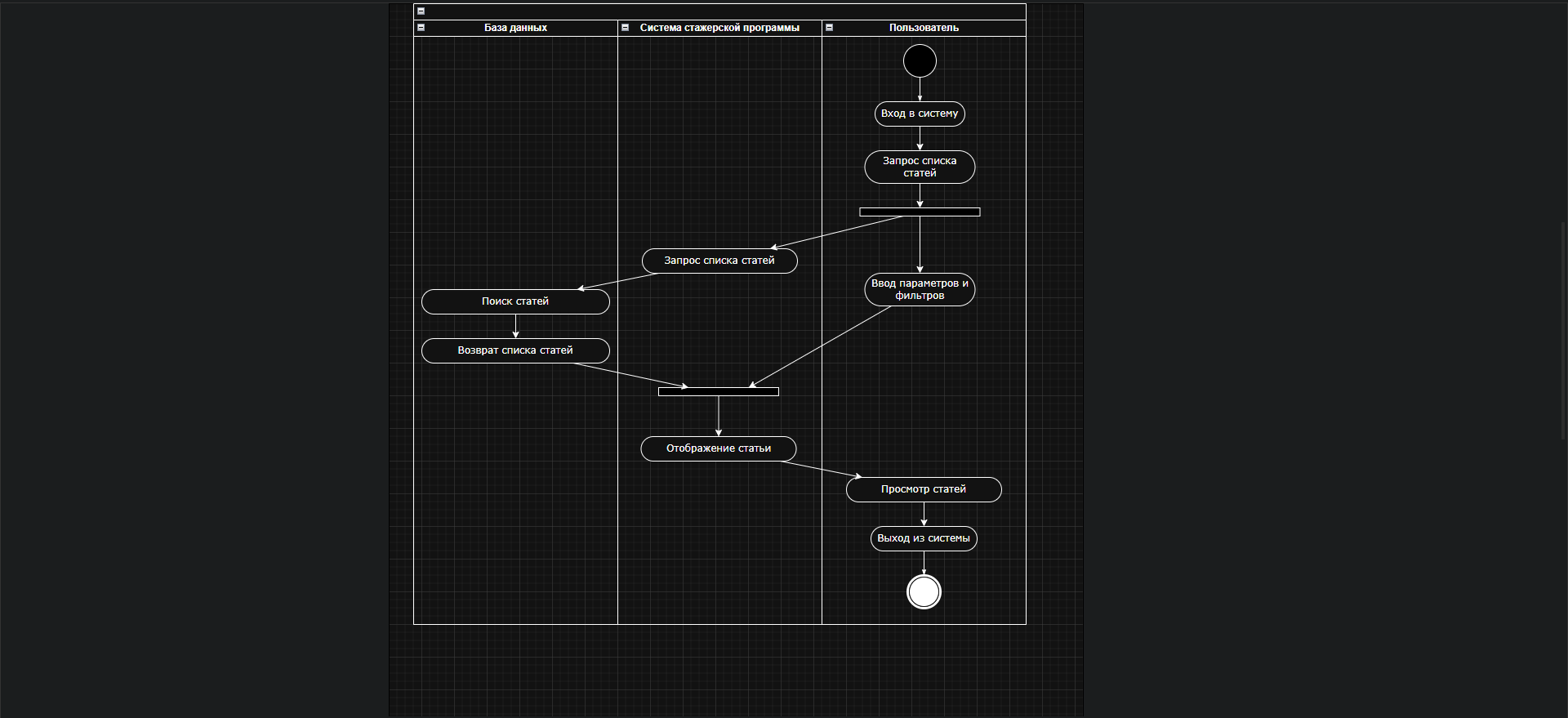
**Рисунок 12. Структура регионального предприятия**

## **ПРИЛОЖЕНИЕ Б**

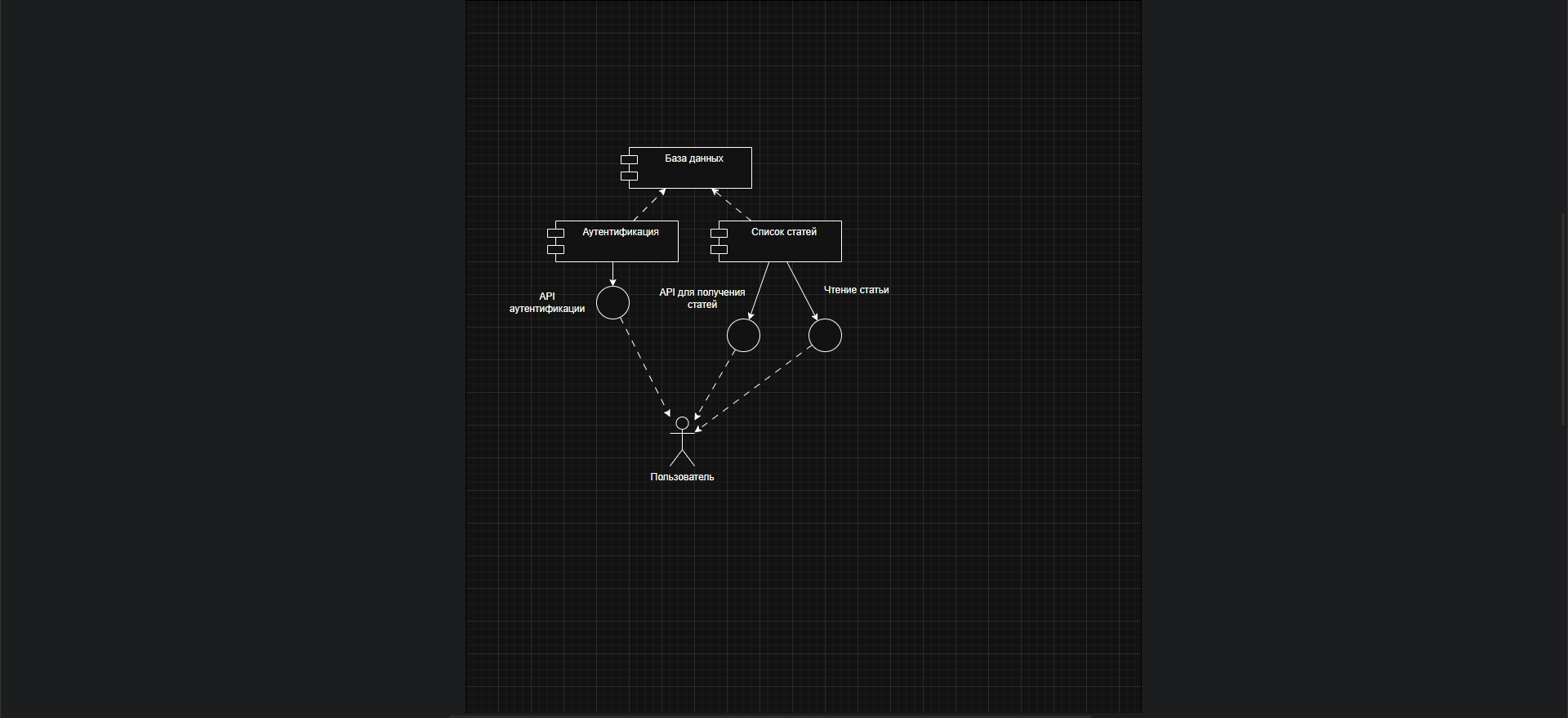
UML-диаграммы модулей программы

****

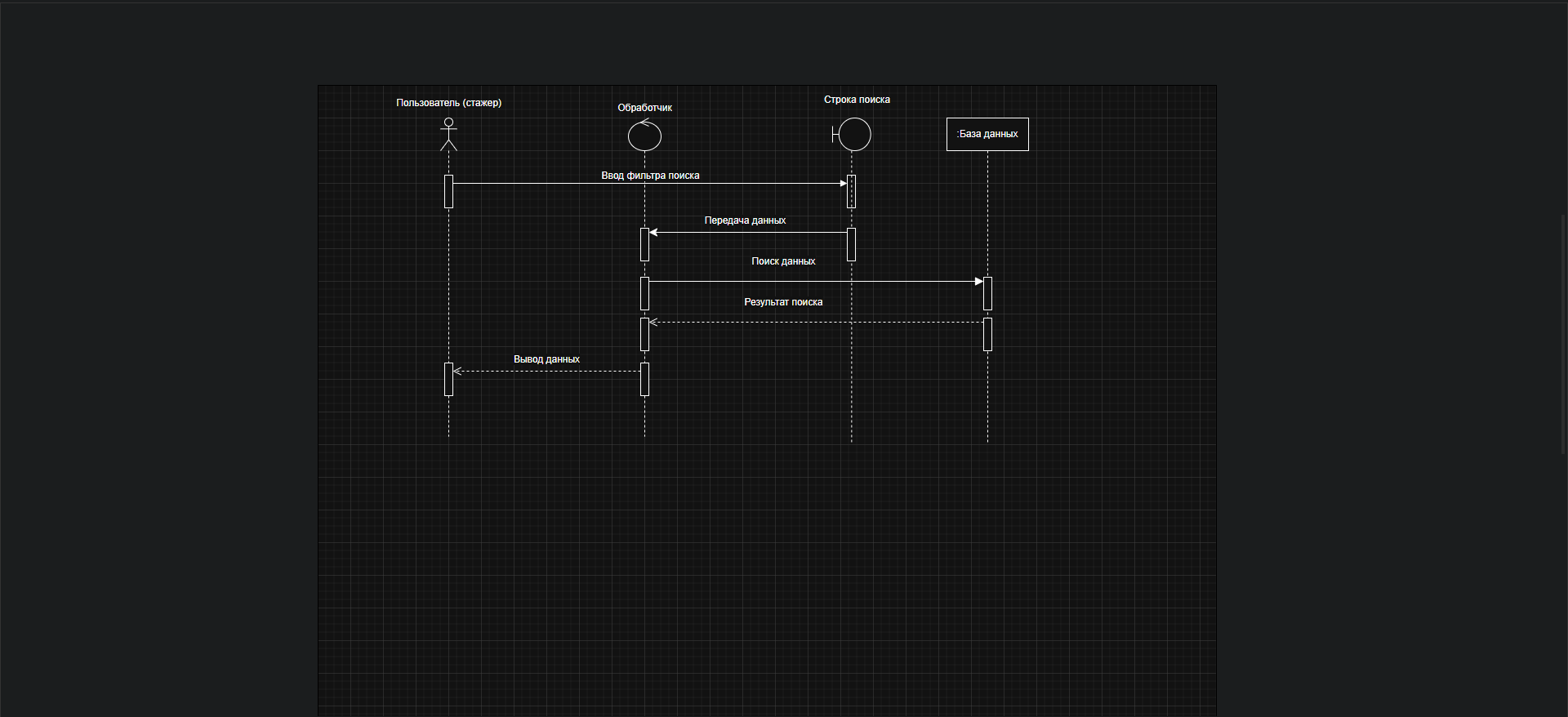
**Рисунок 13. Материалы - диаграмма коопераций**

****

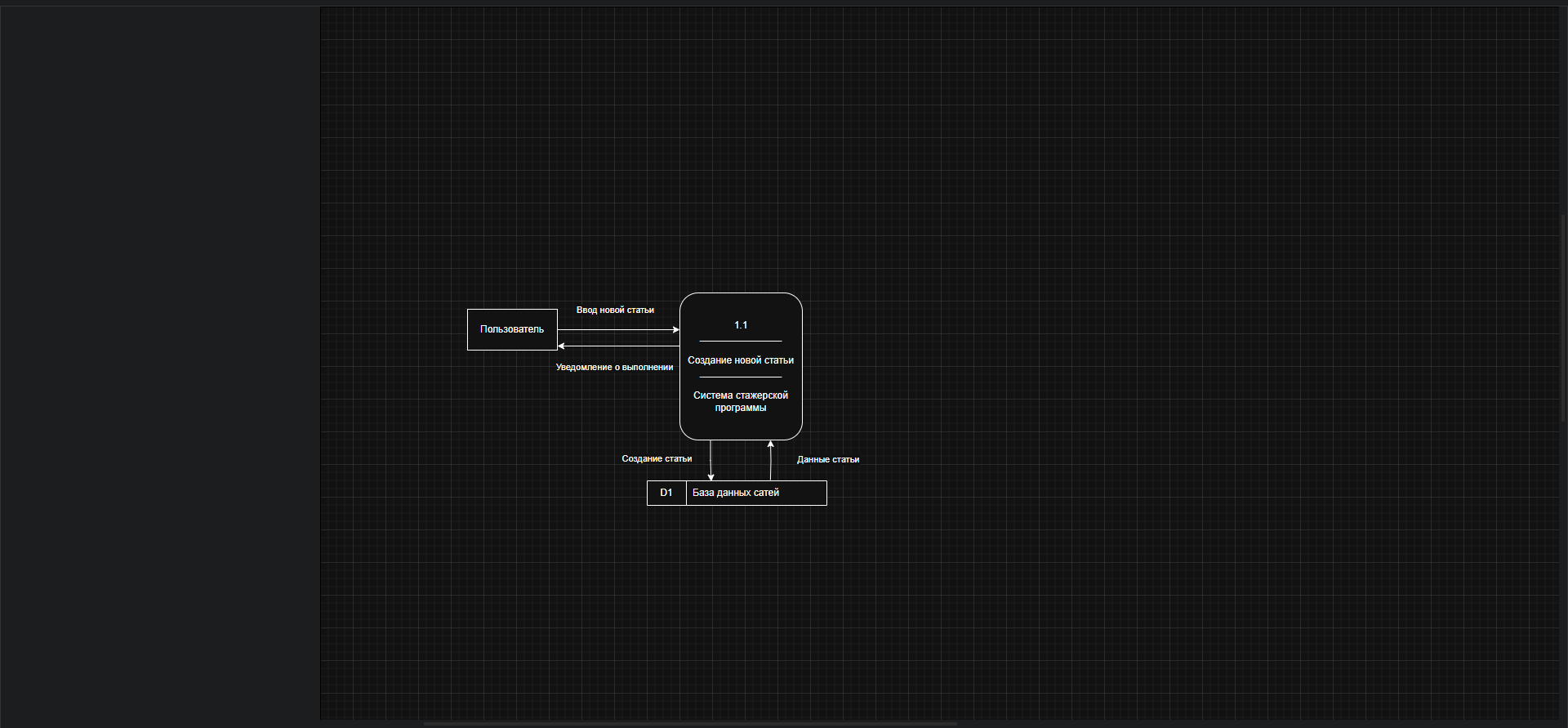
**Рисунок 14. Материалы - диаграмма деятельности**

****

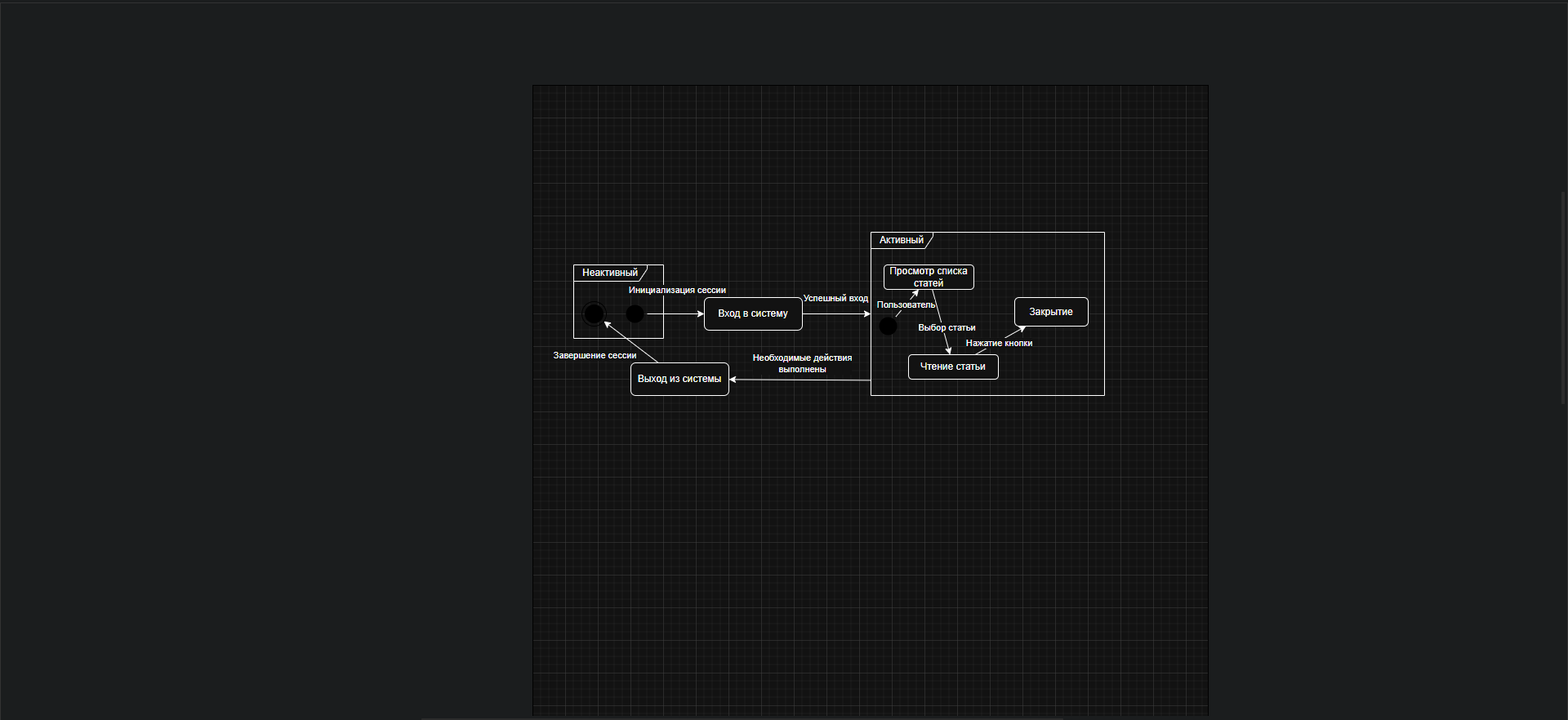
**Рисунок 15. Материалы - диаграмма компонентов**

****

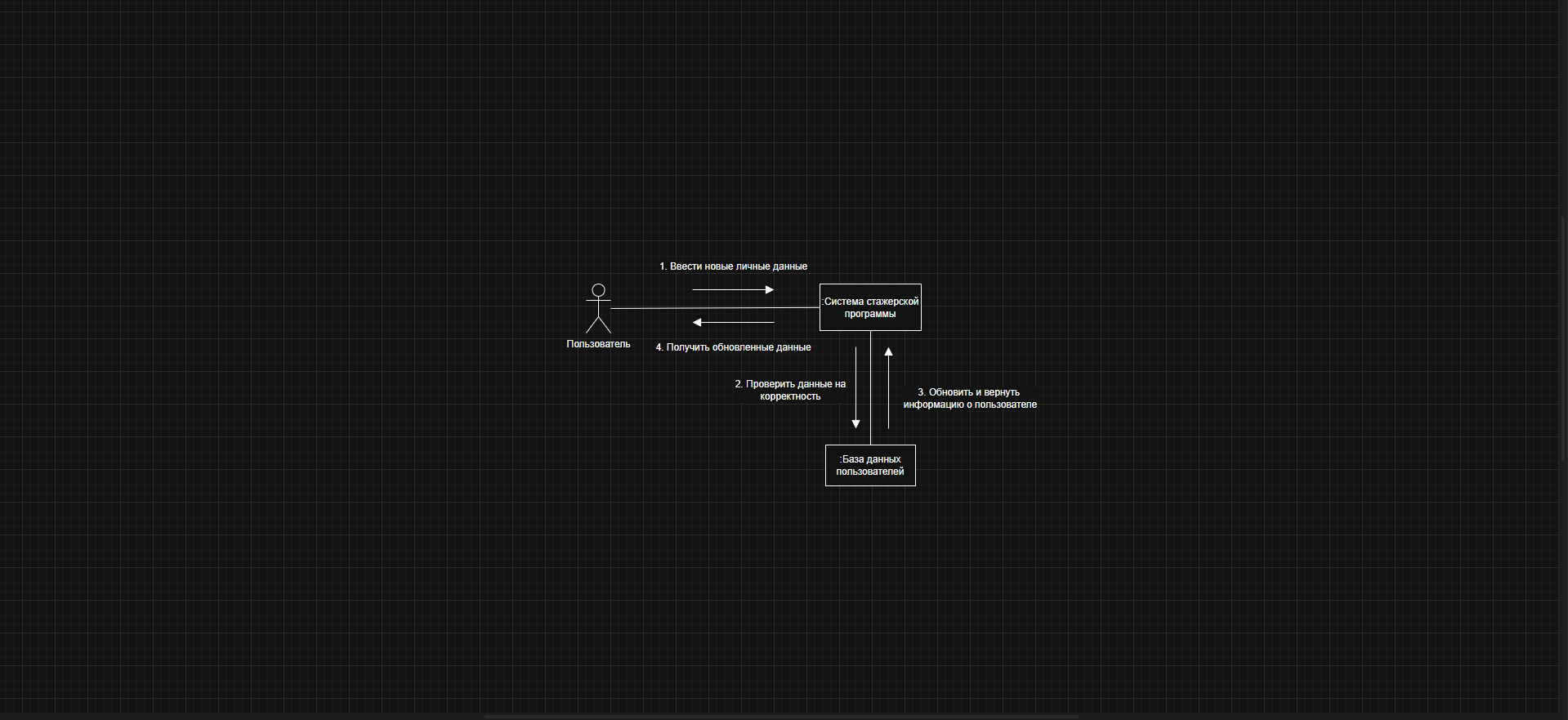
**Рисунок 16. Материалы - диаграмма последовательности**

****

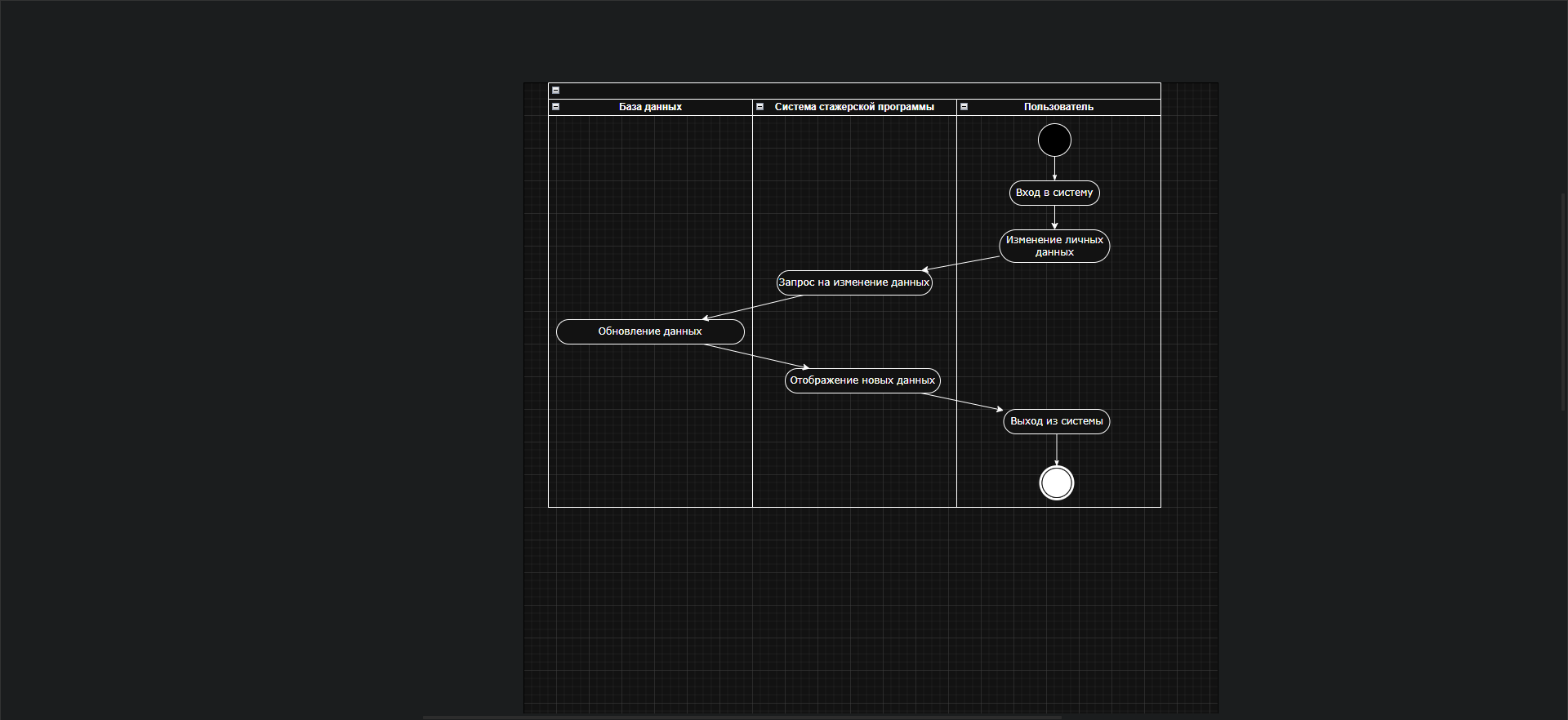
**Рисунок 17. Материалы - диаграмма потоков данных**

****

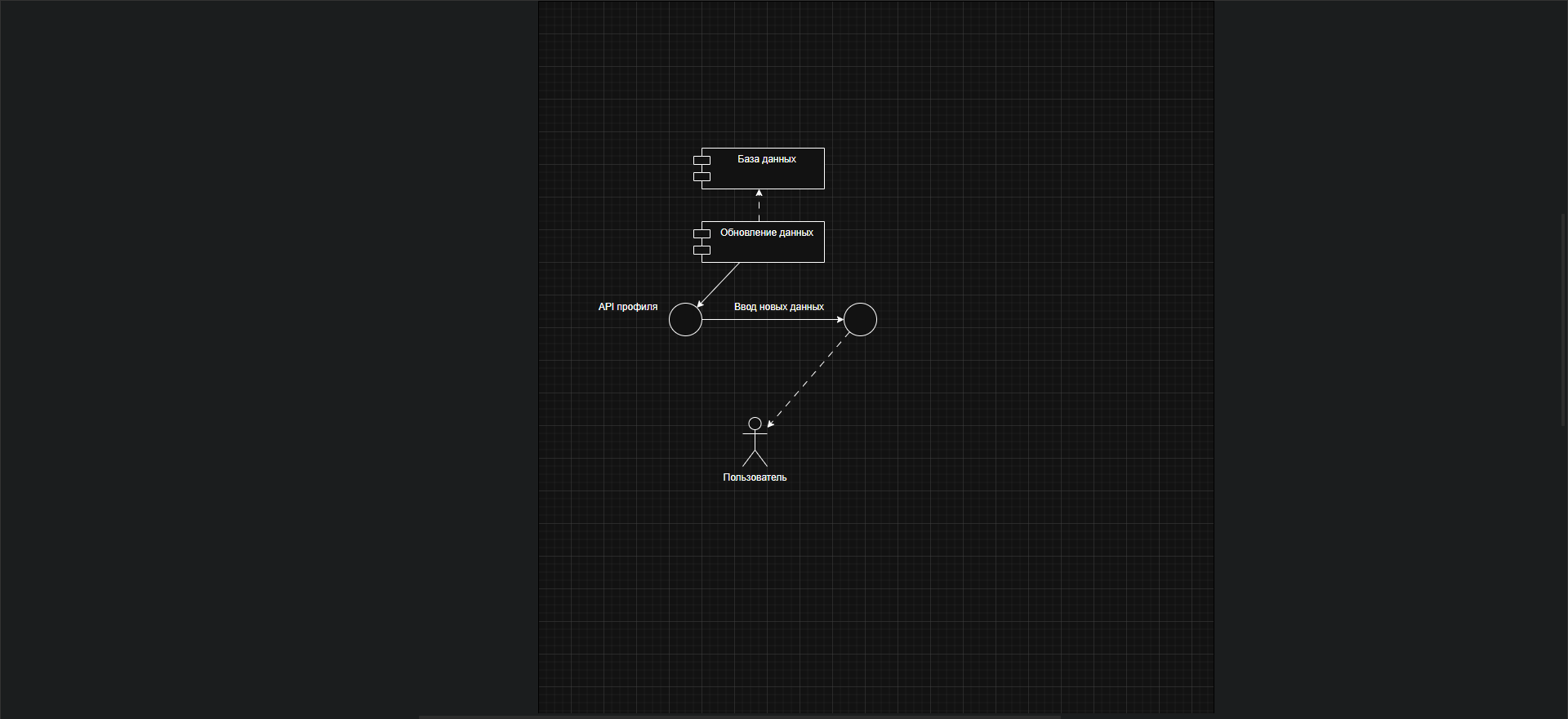
**Рисунок 18. Материалы - диаграмма состояния**

****

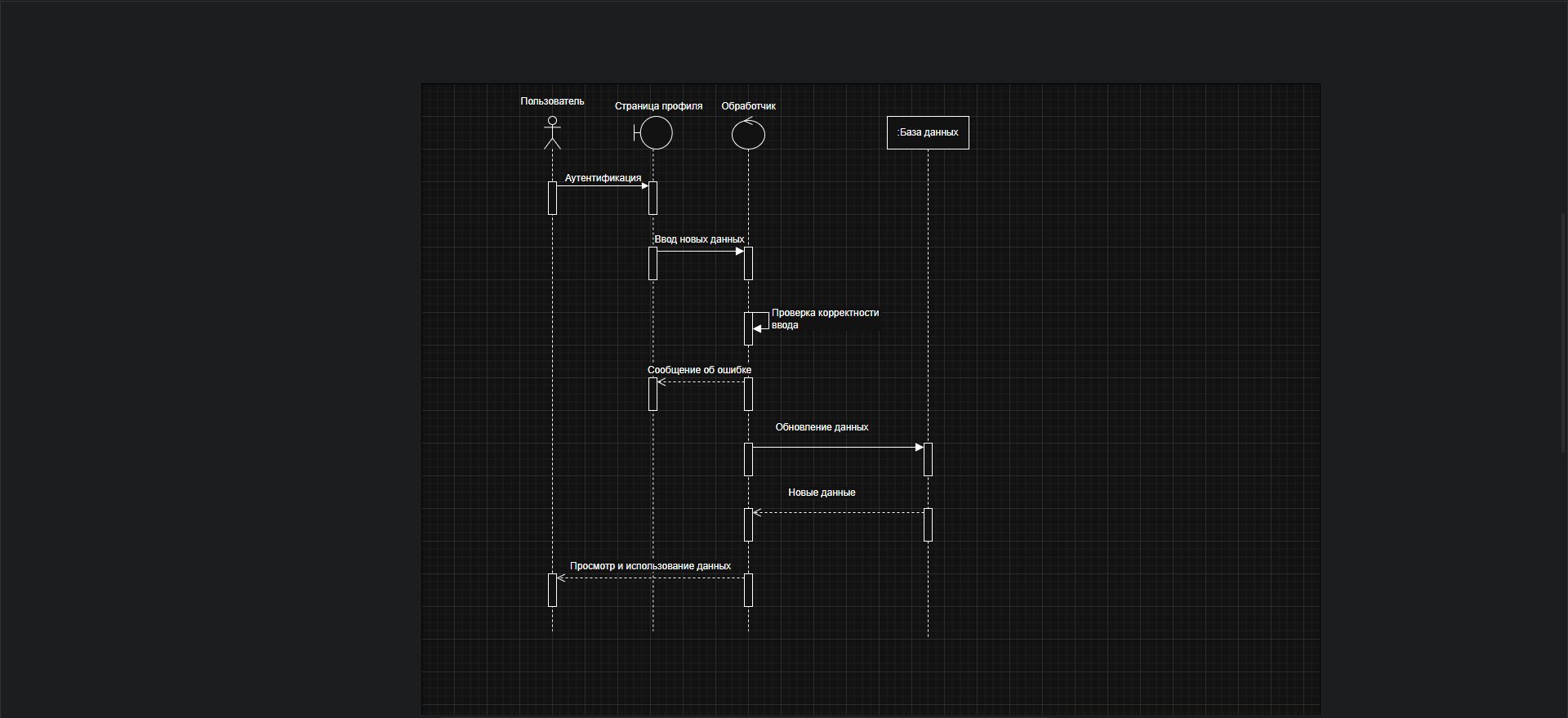
**Рисунок 19. Обновление данных - диаграмма кооперации**

****

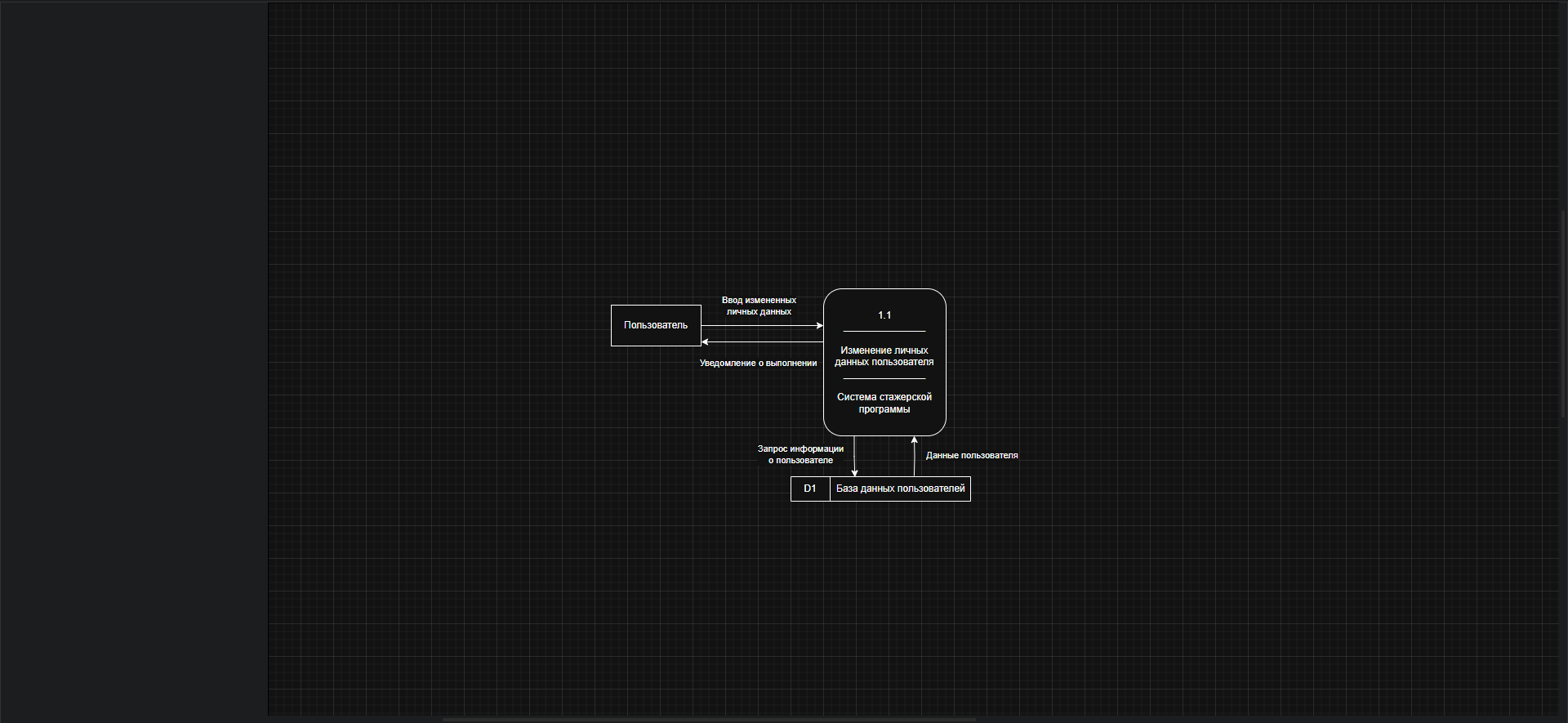
**Рисунок 20. Обновление данных - диаграмма деятельности**

****

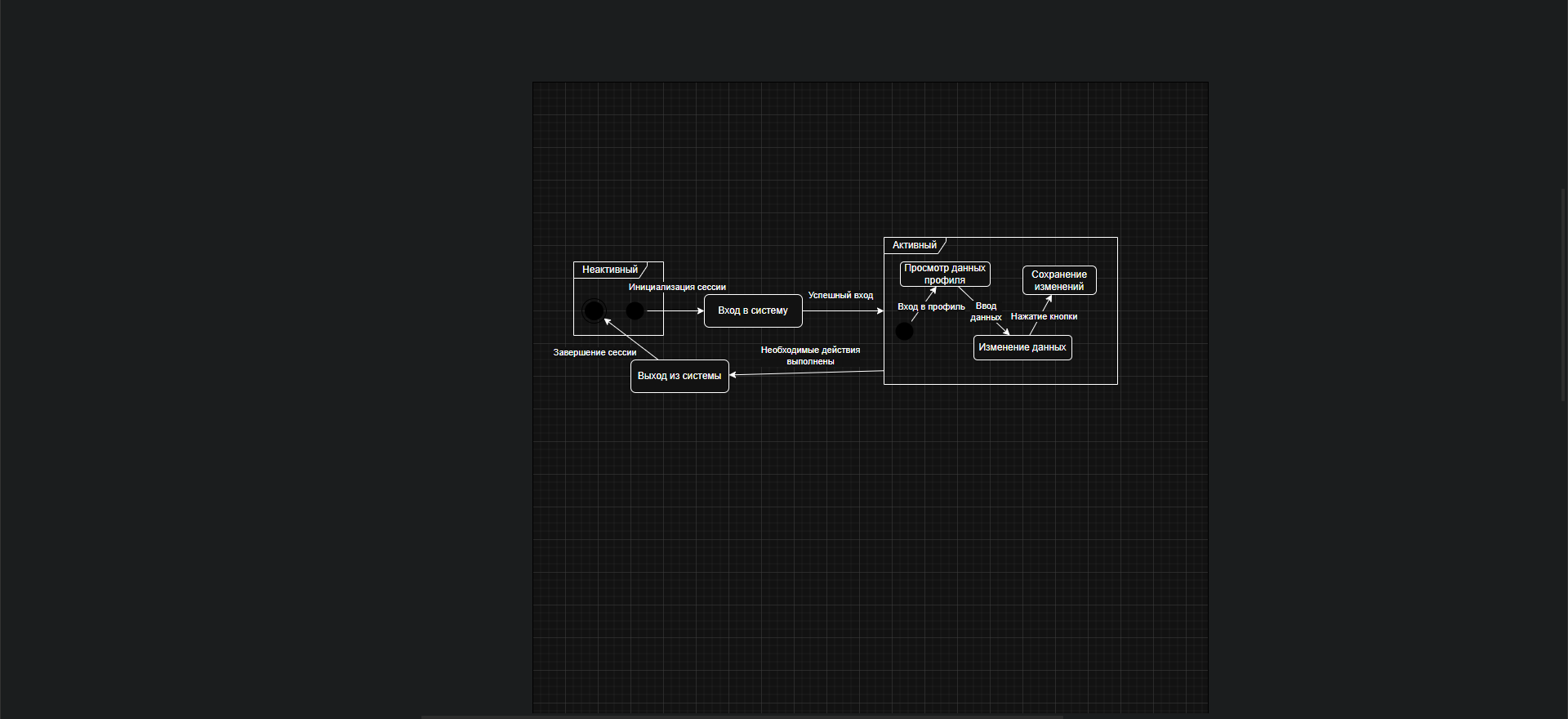
**Рисунок 21. Обновление данных - диаграмма компонентов**

****

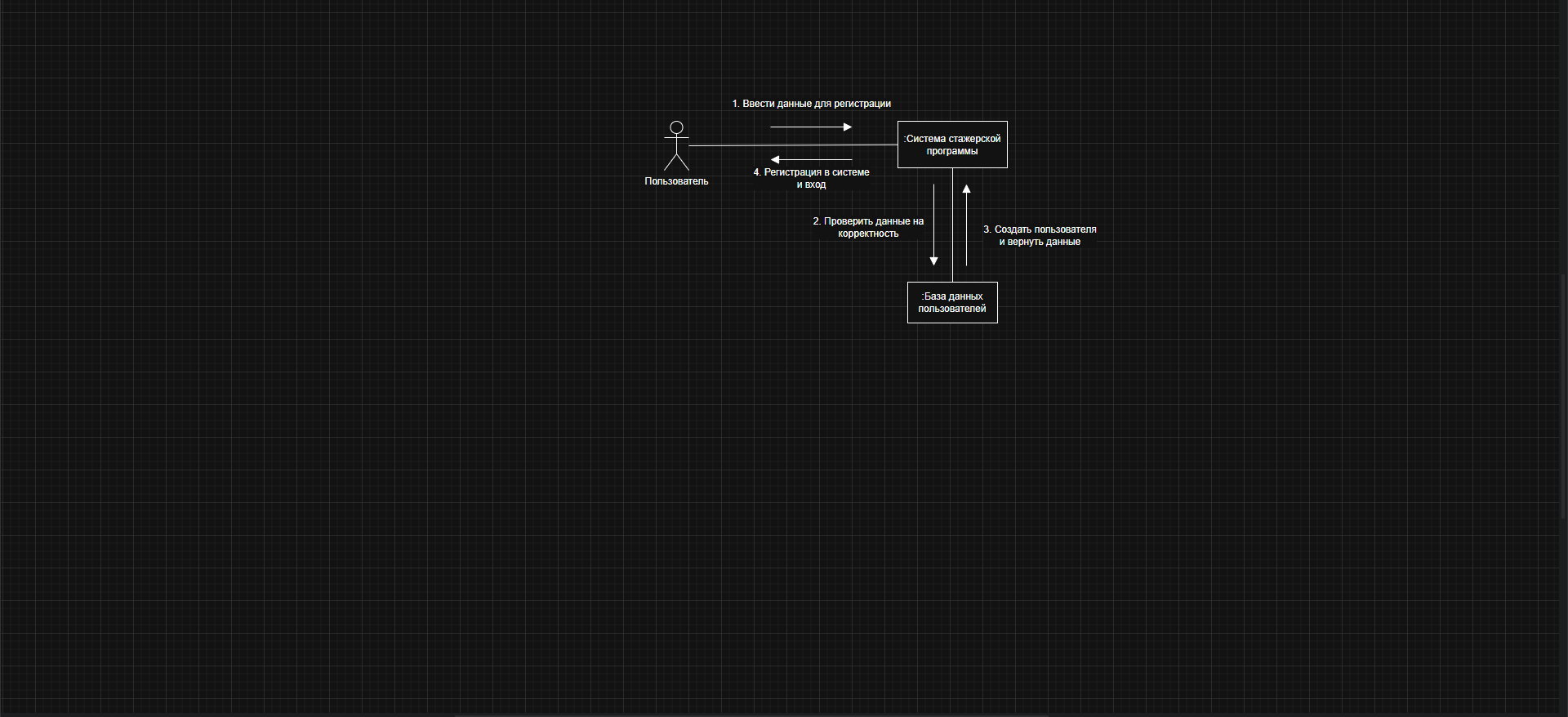
**Рисунок 22. Обновление данных - диаграмма последовательности**

****

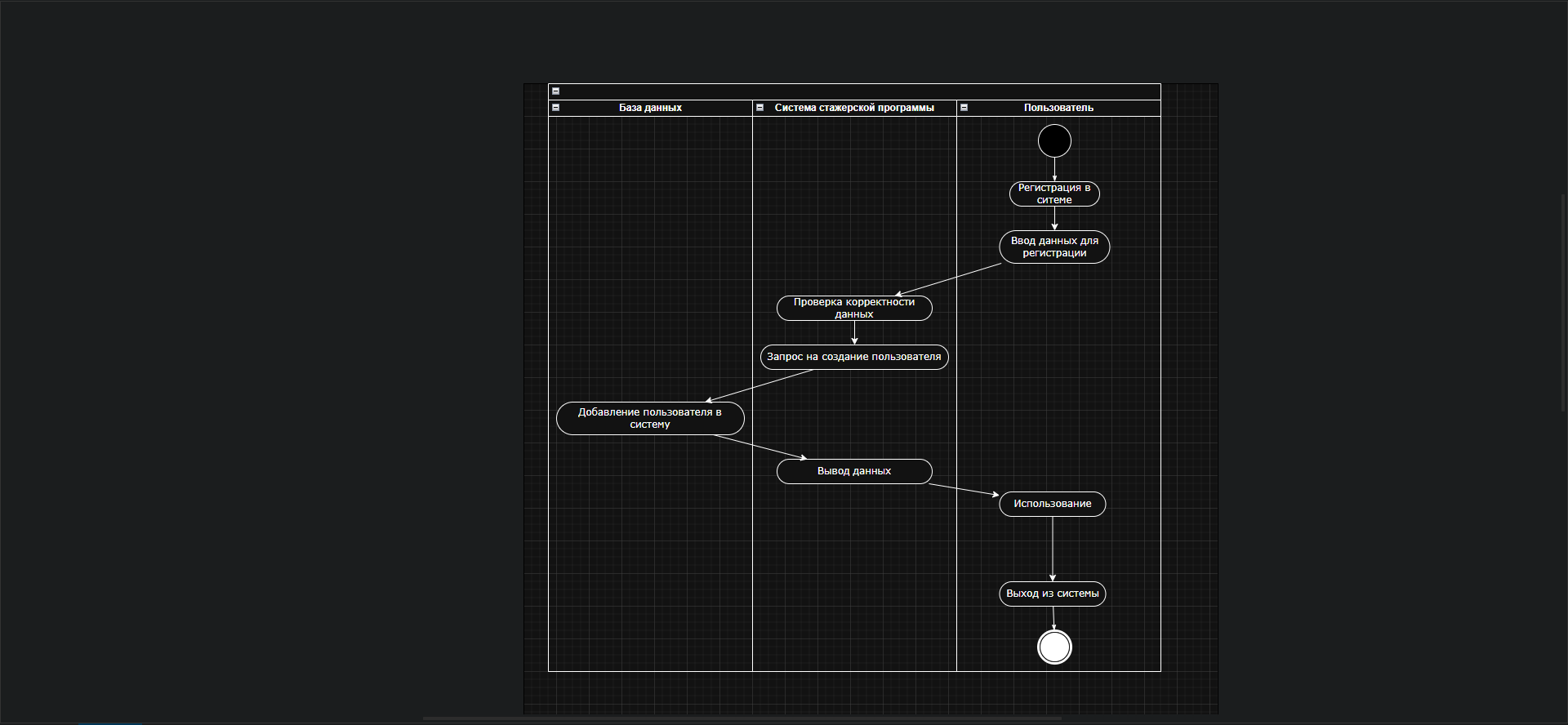
**Рисунок 23. Обновление данных - диаграмма потоков данных**

****

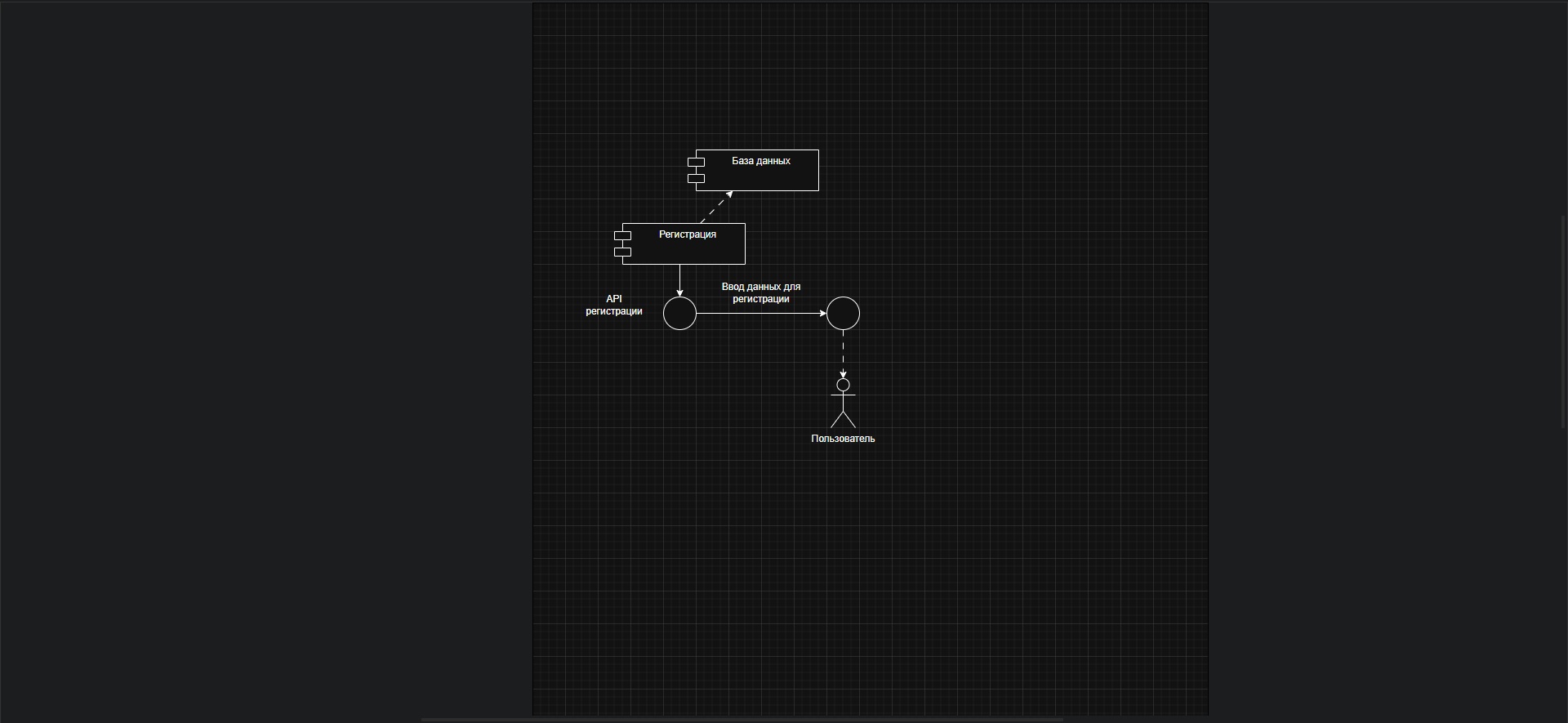
**Рисунок 24. Обновление данных - диаграмма состояния**

****

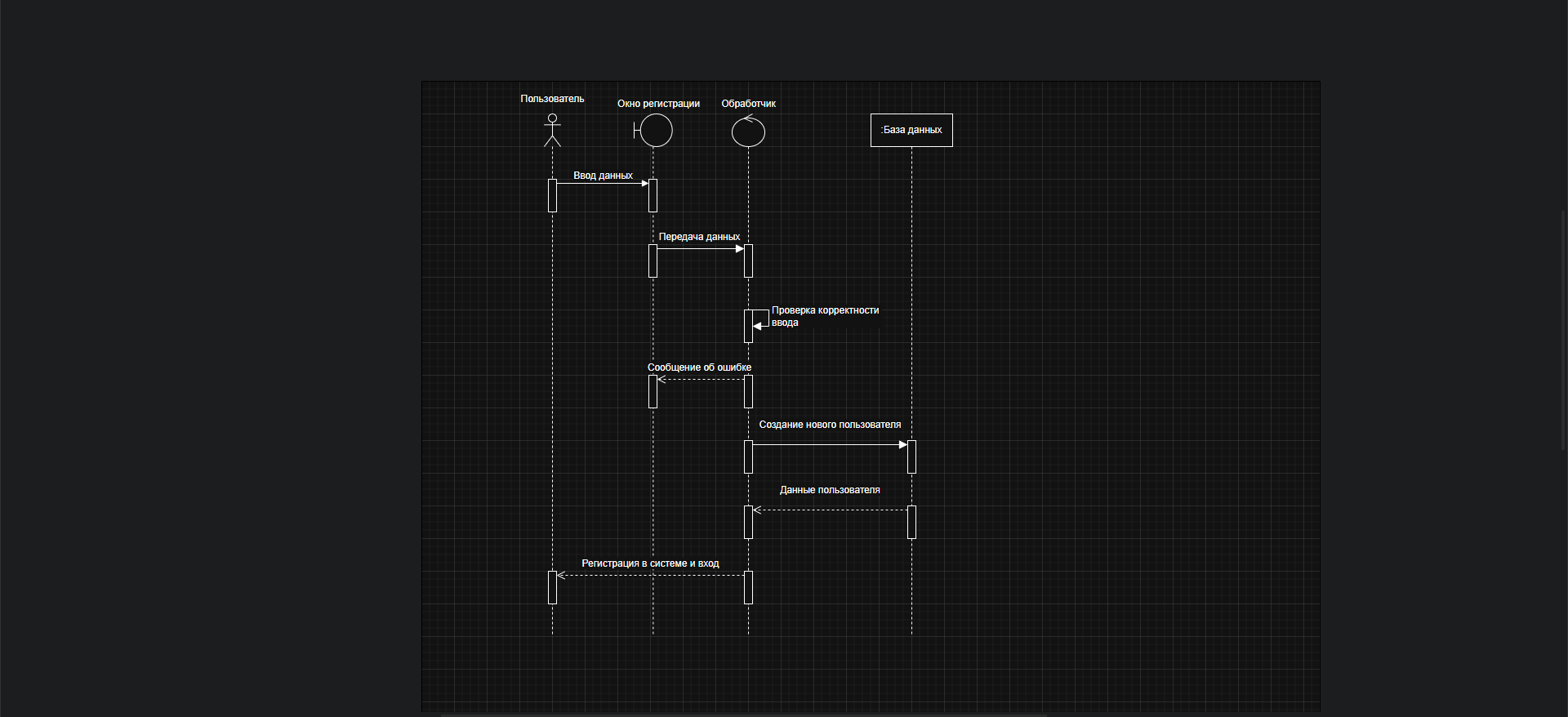
**Рисунок 25. Регистрация - диаграмма кооперации**

****

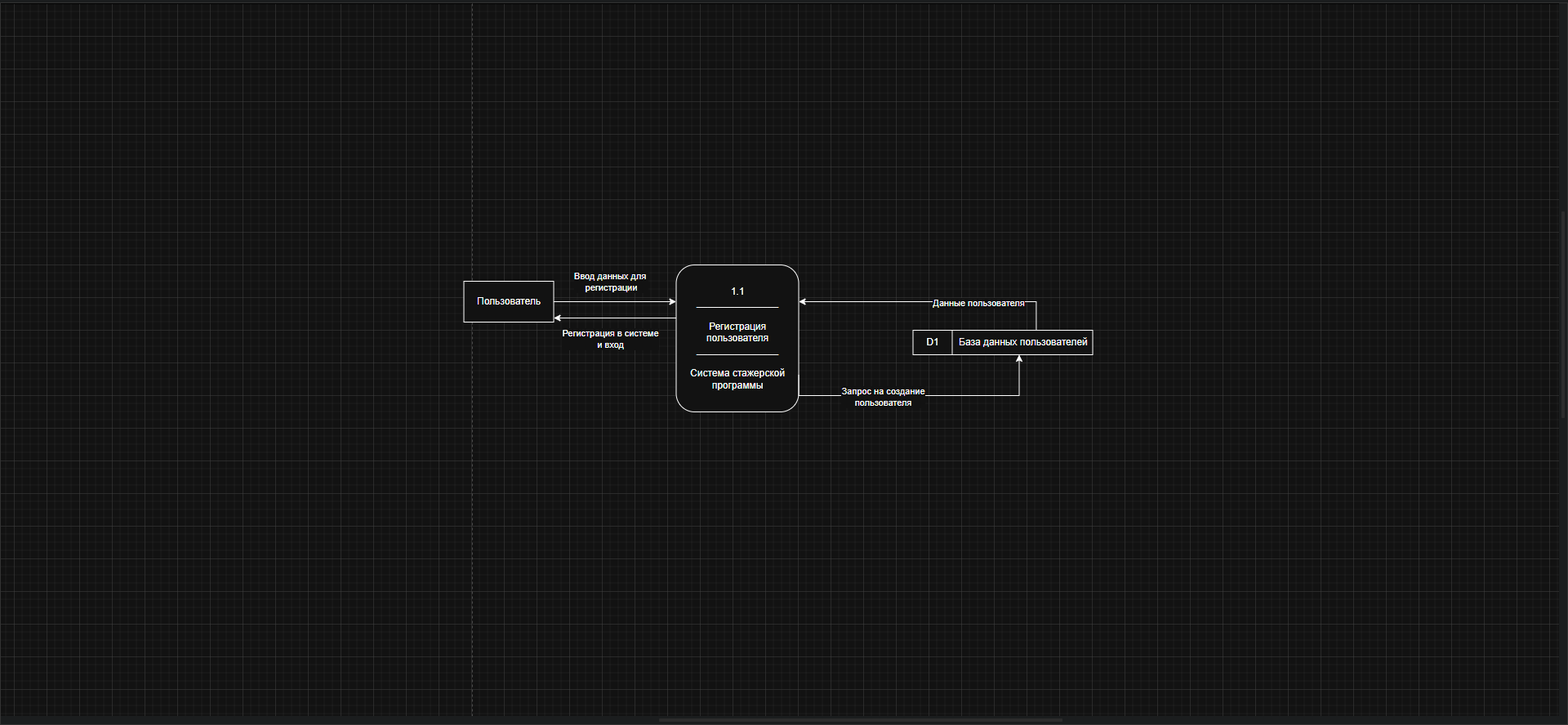
**Рисунок 26. Регистрация - диаграмма деятельности**

****

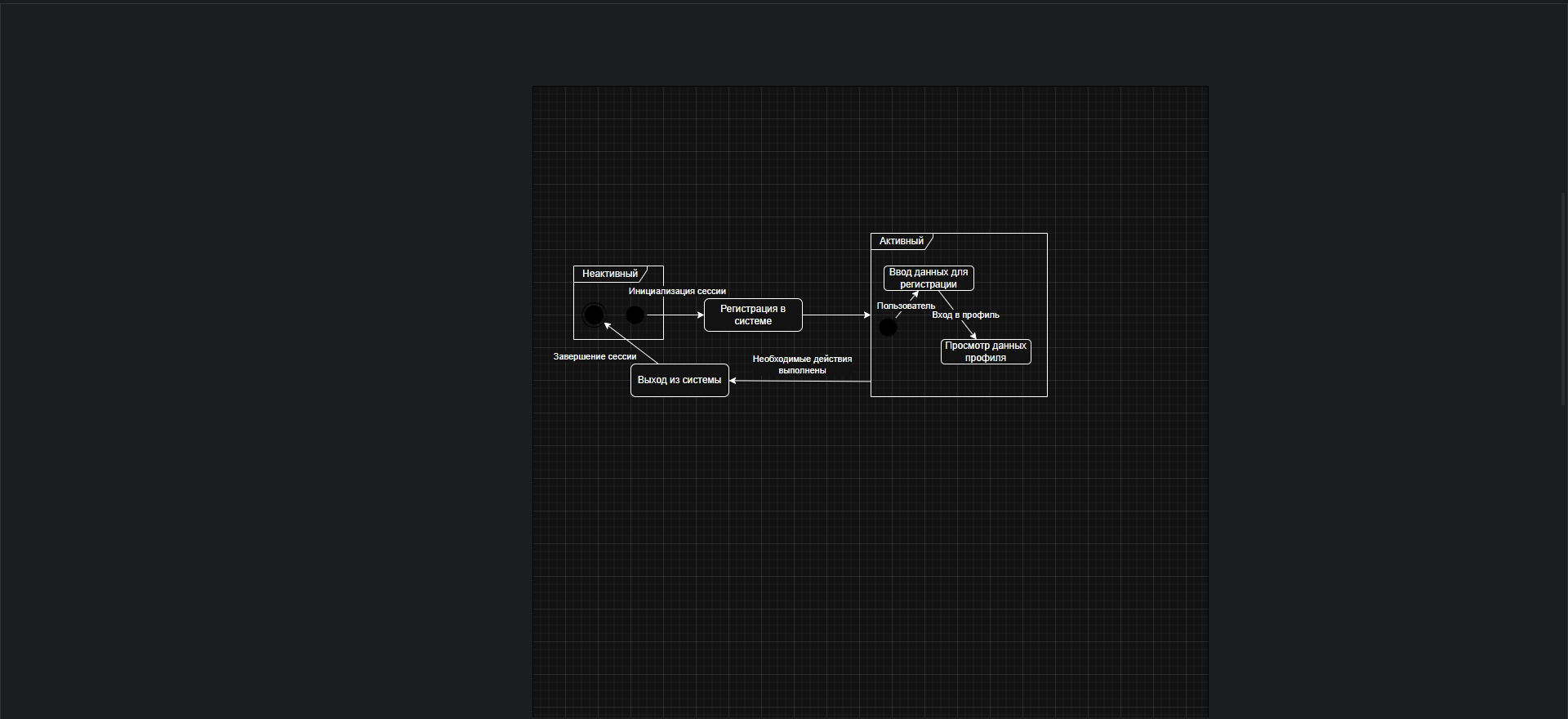
**Рисунок 27. Регистрация - диаграмма компонентов**

****

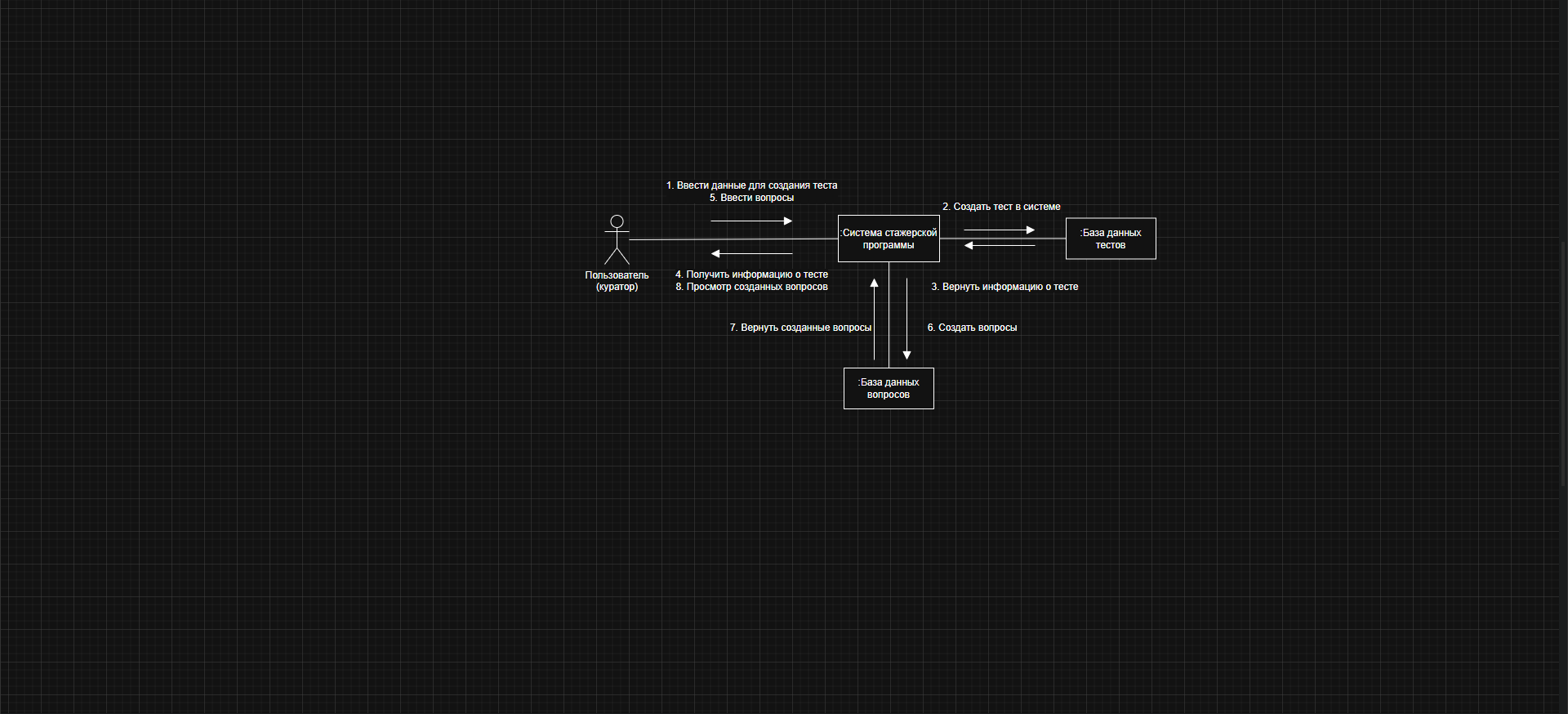
**Рисунок 28. Регистрация - диаграмма последовательности**

****

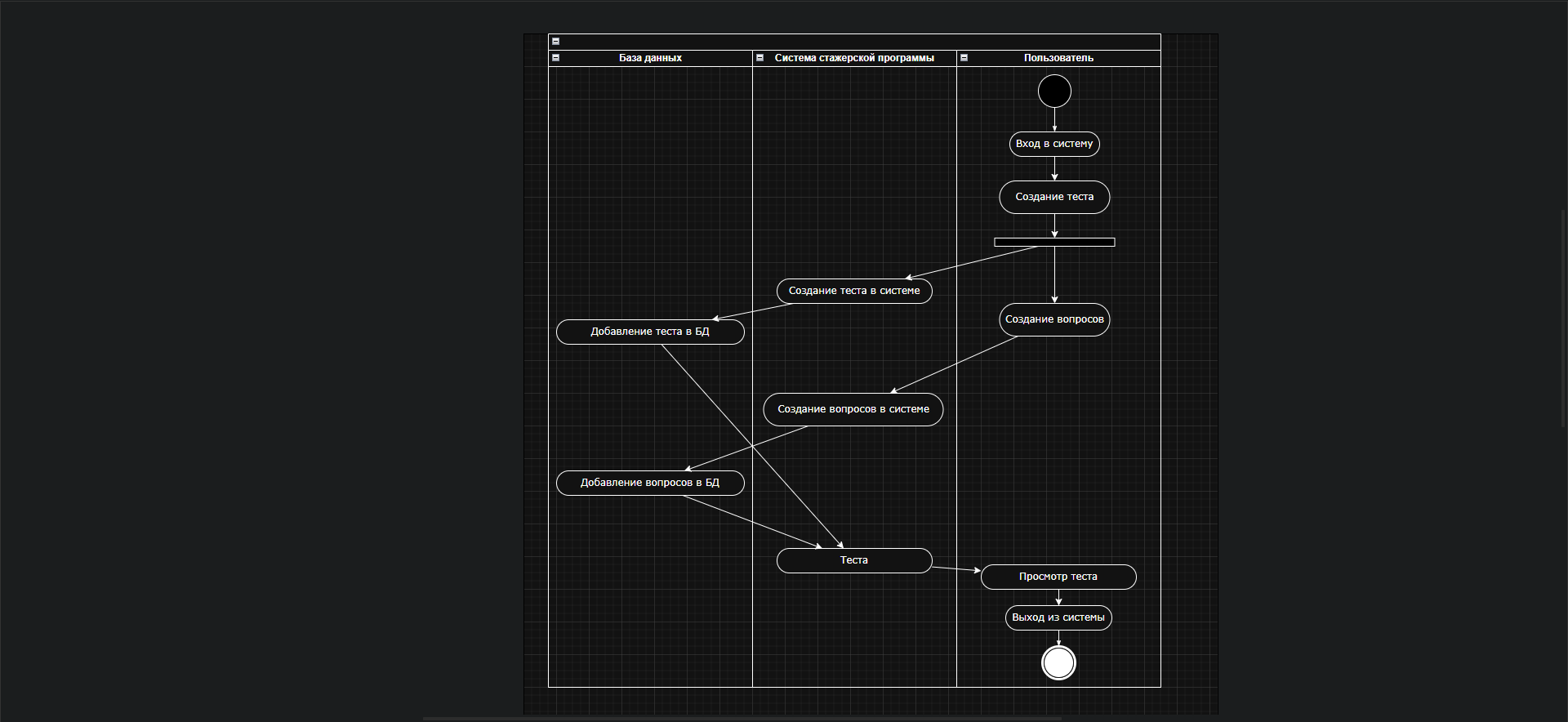
**Рисунок 29. Регистрация - диаграмма потоков данных**

****

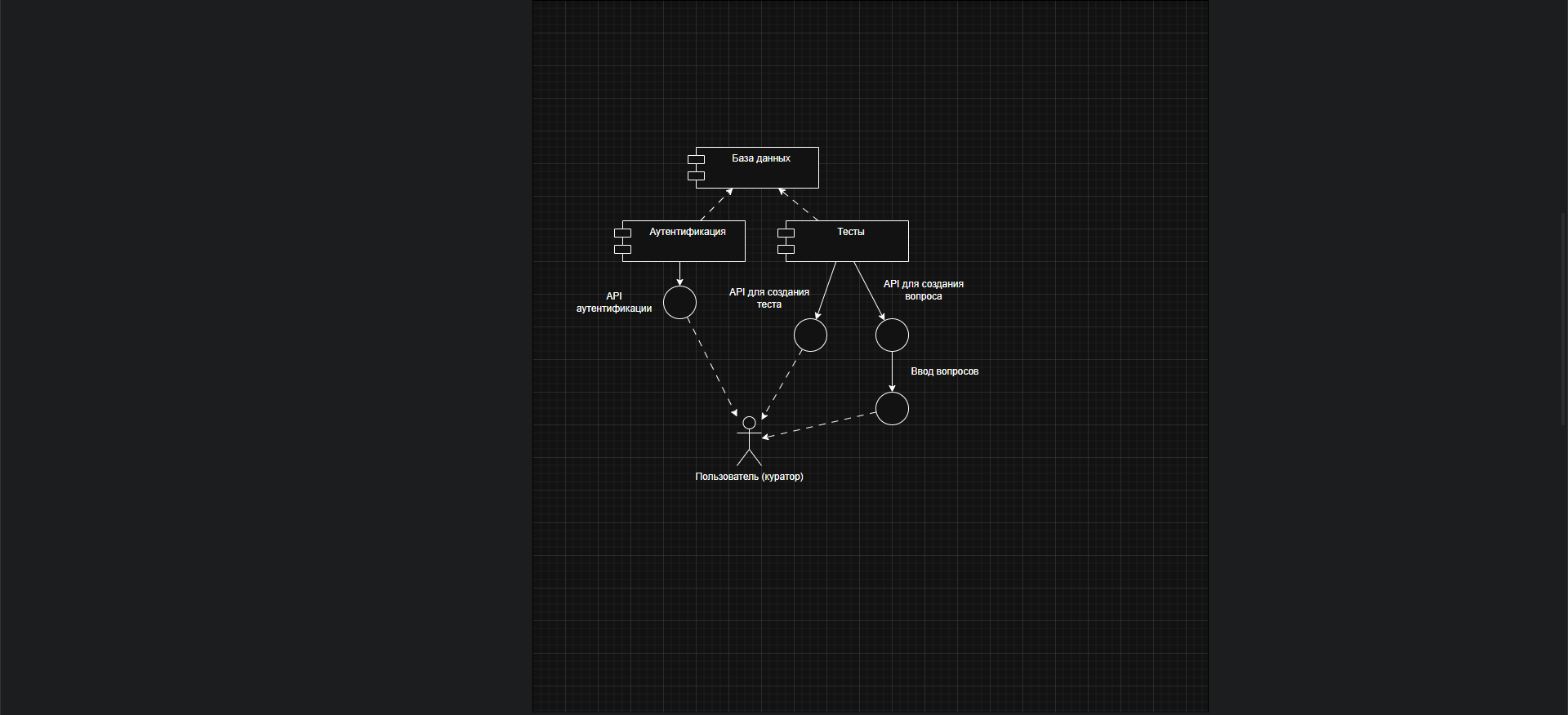
**Рисунок 30. Регистрация - диаграмма состояния**

****

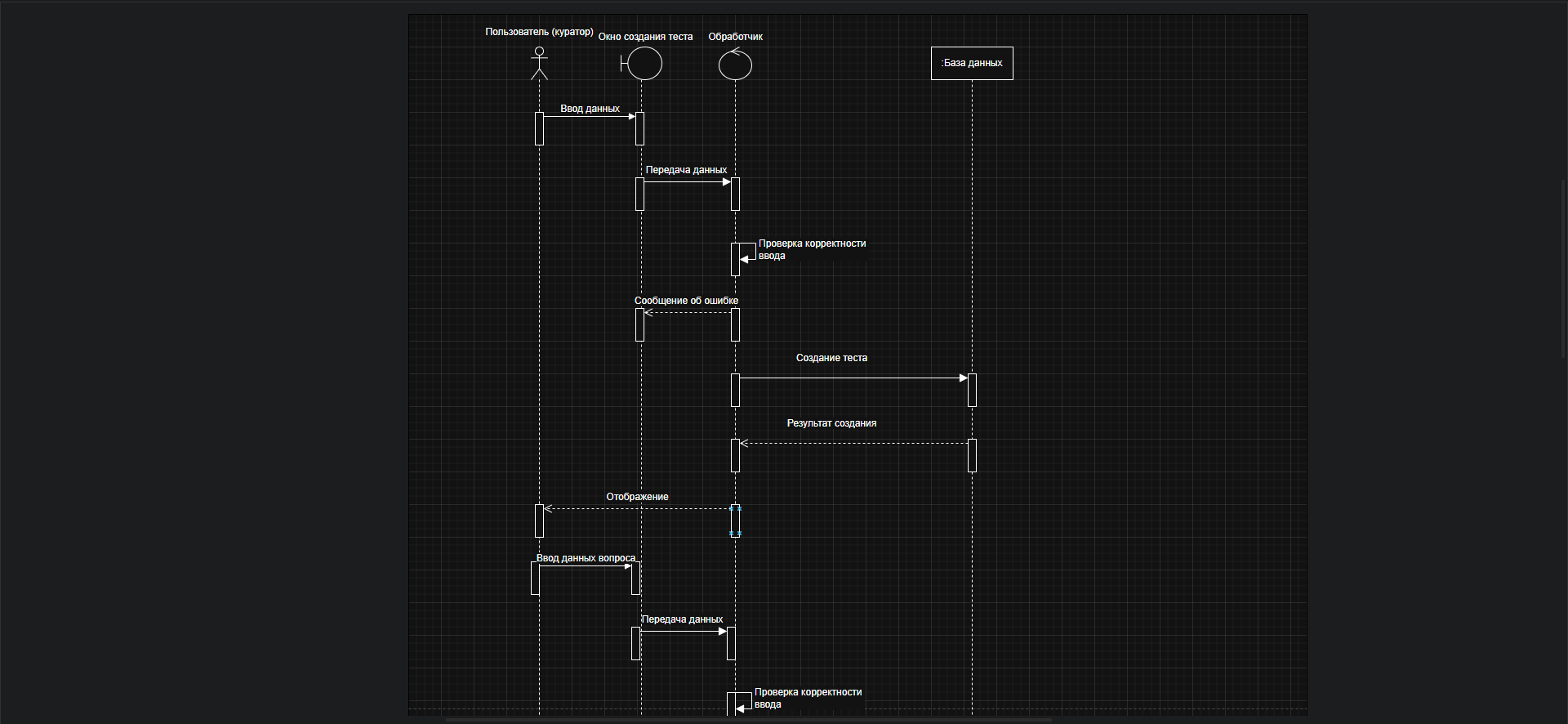
**Рисунок 31. Тесты - диаграмма кооперации**

****

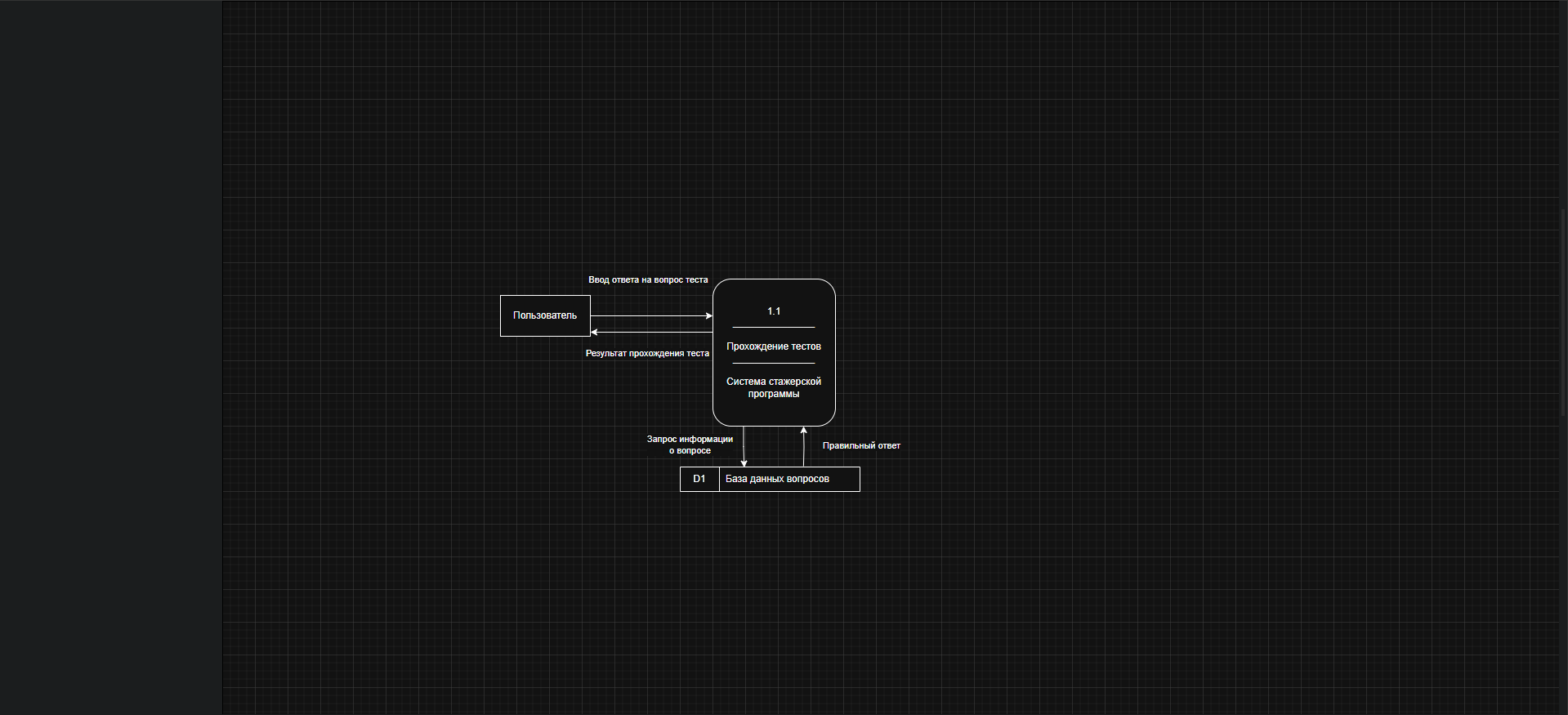
**Рисунок 32. Тесты - диаграмма деятельности**

****

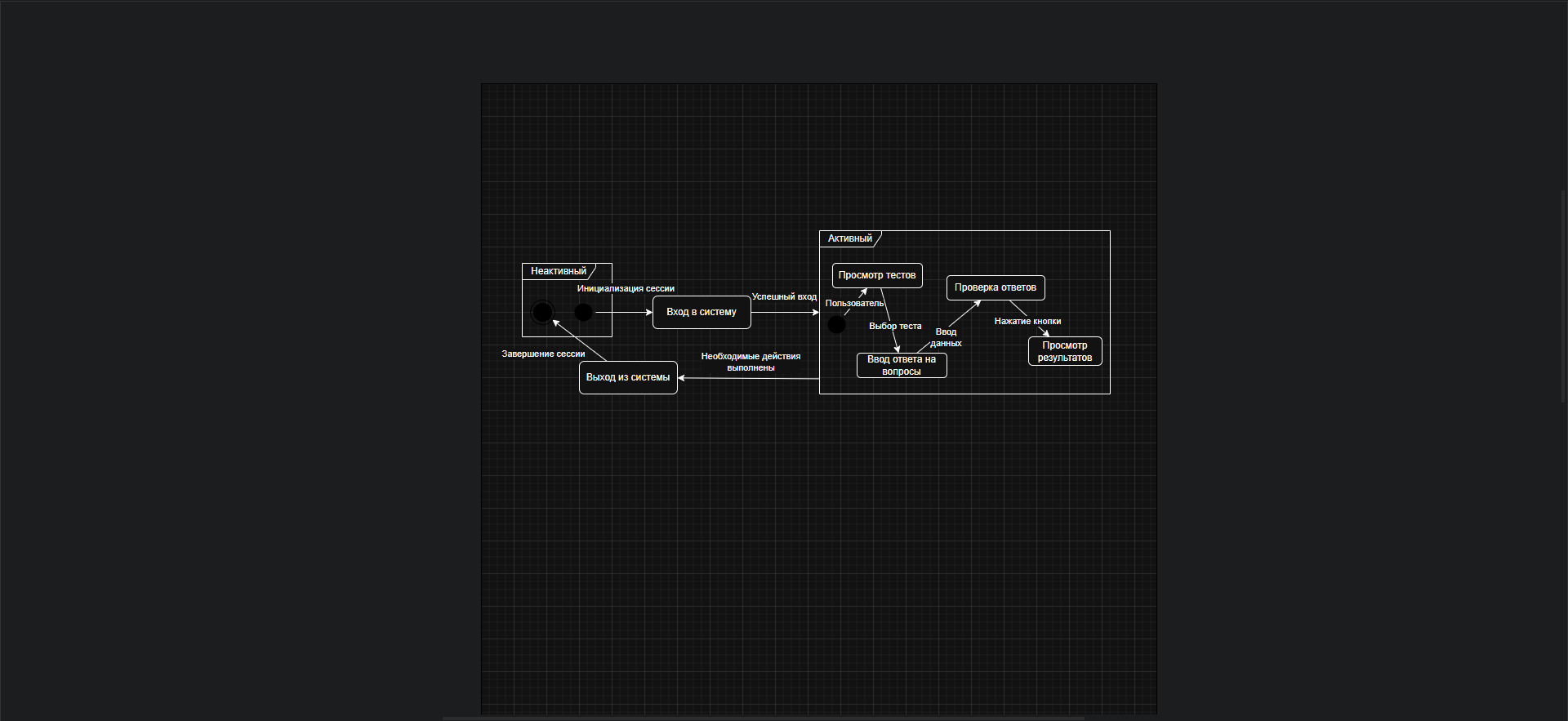
**Рисунок 33. Тесты - диаграмма компонентов**

****

**Рисунок 34. Тесты - диаграмма последовательности**

****

**Рисунок 35. Тесты - диаграмма потоков данных**

****

**Рисунок 36. Тесты - диаграмма состояния**

## **ПРИЛОЖЕНИЕ В**