Государственное бюджетное профессиональное образовательное учреждение

«Нижегородский Губернский колледж»

**ОТЧЕТ**

**ПО ВЫПОЛНЕНИЮ ЗАДАНИЯ ПРОИЗВОДСТВЕННОЙ ПРАКТИКИ**

ПМ.02 ОСУЩЕСТВЛЕНИЕ ИНТЕГРАЦИИ ПРОГРАММНЫХ МОДУЛЕЙ

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

Нижний Новгород

2025

**СОДЕРЖАНИЕ**

[**ВВЕДЕНИЕ** 3](#_Toc201573357)

[**ТЕОРЕТИЧЕСКАЯ ЧАСТЬ** 5](#_Toc201573358)

[**1.1 Организационная структура предприятия** 5](#_Toc201573359)

[**1.2 Анализ программного обеспечения и технического оснащения** 6](#_Toc201573360)

[**1.3 Описание инструментальных средств разработки и сопровождения программных средств** 8](#_Toc201573361)

[**ПРАКТИЧЕСКАЯ ЧАСТЬ** 9](#_Toc201573362)

[**2.1 Анализ предметной области** 9](#_Toc201573363)

[**2.2 Оформление технического задания** 11](#_Toc201573364)

[**2.3 Проектирование программного продукта** 13](#_Toc201573365)

[**2.4 Разработка программного модуля** 20](#_Toc201573366)

[**2.5 Тестирование программного продукта** 22](#_Toc201573367)

[**2.6 Разработка модуля математического моделирования** 24](#_Toc201573368)

[**ЗАКЛЮЧЕНИЕ** 25](#_Toc201573369)

[**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ** 26](#_Toc201573370)

[**ПРИЛОЖЕНИЯ** 27](#_Toc201573371)

[**ПРИЛОЖЕНИЕ А** 27](#_Toc201573372)

[**ПРИЛОЖЕНИЕ Б** 28](#_Toc201573373)

[**ПРИЛОЖЕНИЕ В** 36](#_Toc201573374)

# **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 контактов.

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

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

# **ПРАКТИЧЕСКАЯ ЧАСТЬ**

## **2.1 Анализ предметной области**

На начальной стадии создания платформы для адаптации стажёров был проведен глубокий анализ предметной области. Эта работа стала отправной точкой для всего проекта, позволив сформировать четкое понимание того, какой должна быть система.

Основные этапы работы:

1. **Изучение потребностей пользователей**

Первым делом был проведен тщательный анализ того, какие задачи должна решать система. Для этого:

* Провели опрос среди работников и руководства
* Опросили стажёров прошлых наборов
* Изучили существующие процессы адаптации в компании

В результате были определены ключевые функции:

* Централизованное хранение учебных материалов
* Система проверки знаний
* Мониторинг прогресса обучения

1. **Определение структуры данных**

На этом этапе были выявлены основные сущности системы и связи между ними:

* Профили пользователей (стажёры, кураторы, администраторы)
* Учебные курсы и программы
* Тестовые задания и результаты
* Отчеты и аналитика

Для каждой сущности прописали важные атрибуты и правила взаимодействия.

1. **Создание концептуальной модели**

Заключительный этап включал:

* Визуализацию структуры системы
* Формализацию взаимосвязей между компонентами
* Разработку принципов организации данных

Проделанная работа позволила:

1. Чётко очертить границы системы
2. Выделить все значимые элементы и их характеристики
3. Установить логические связи между компонентами
4. Сформировать единое видение системы
5. Создать основу для последующей технической реализации

Полученная концептуальная модель стала надёжным фундаментом для дальнейшей разработки. Она помогает избежать многих проблем на этапе программирования, так как все ключевые решения уже продуманы и согласованы. Имея на руках результаты этого анализа, мы можем переходить к детальному проектированию архитектуры системы, разработке интерфейсов и планированию этапов реализации.

## **2.2 Оформление технического задания**

Разработка технического задания для платформы адаптации стажёров Intern Guide потребовала комплексного подхода, учитывающего специфику корпоративного обучения в страховой отрасли. Этот документ стал ключевым ориентиром для разработки, обеспечив единое видение системы и её функциональных возможностей.

Формирование технического задания началось с всестороннего исследования потребностей СПАО «Ингосстрах». Был проведён:

1. Анализ существующих процессов адаптации новых сотрудников в компании
2. Изучение современных LMS-систем
3. Опрос сотрудников

Это позволило выделить критические требования к системе:

* Интуитивно понятный интерфейс для пользователей с разным уровнем технической подготовки
* Гибкая система управления учебными материалами
* Инструменты автоматизированной проверки знаний
* Подробная аналитика успеваемости

Техническое задание для Intern Guide включает несколько взаимосвязанных разделов:

1. Бизнес-требования

* Описание целей внедрения системы
* Ключевые показатели эффективности
* Ожидаемое влияние на бизнес-процессы

1. Функциональные требования:

* Детализированные сценарии использования
* Описание ролей и их прав
* Логика работы основных модулей

1. Технические аспекты:

* Архитектурные решения
* Спецификации интеграции с корпоративными системами

1. Критерии приемки:

* Методики тестирования
* Показатели производительности

При создании технического задания особый акцент был сделан на гибкость системы, её масштабируемость, безопасность данных пользователей и аппаратных средств.

Итоговый документ представляет собой:

1. Руководство для разработчиков с четкими техническими спецификациями
2. Инструмент контроля для заказчика с измеримыми критериями приемки
3. Дорожную карту для будущего развития системы

Техническое задание для Intern Guide отличается практической ориентированностью – все требования основаны на реальных бизнес-процессах компании и подкреплены данными исследований.

Этот документ стал надежной основой для создания системы, которая реально решает проблемы адаптации новых сотрудников в крупной страховой компании.

## **2.3 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 9 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**

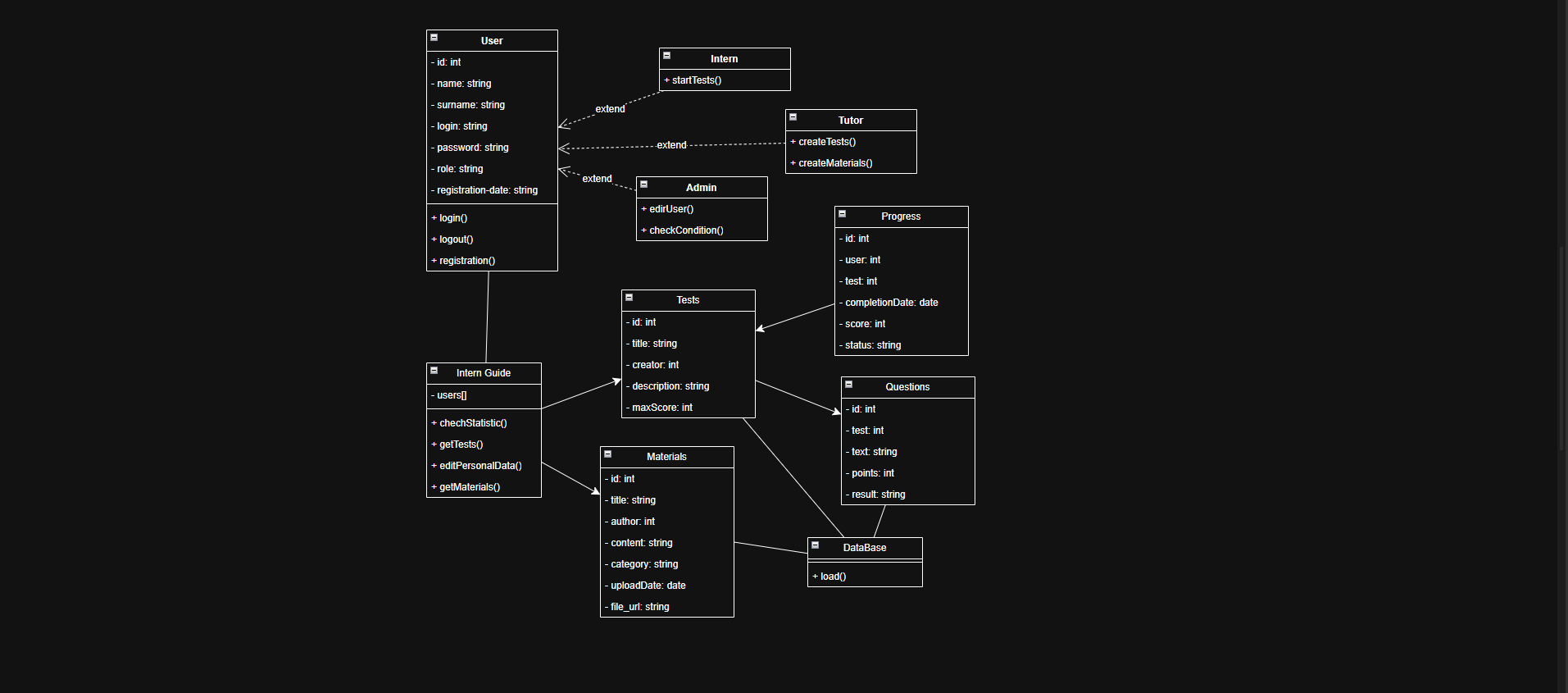
Тип: Структурная

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

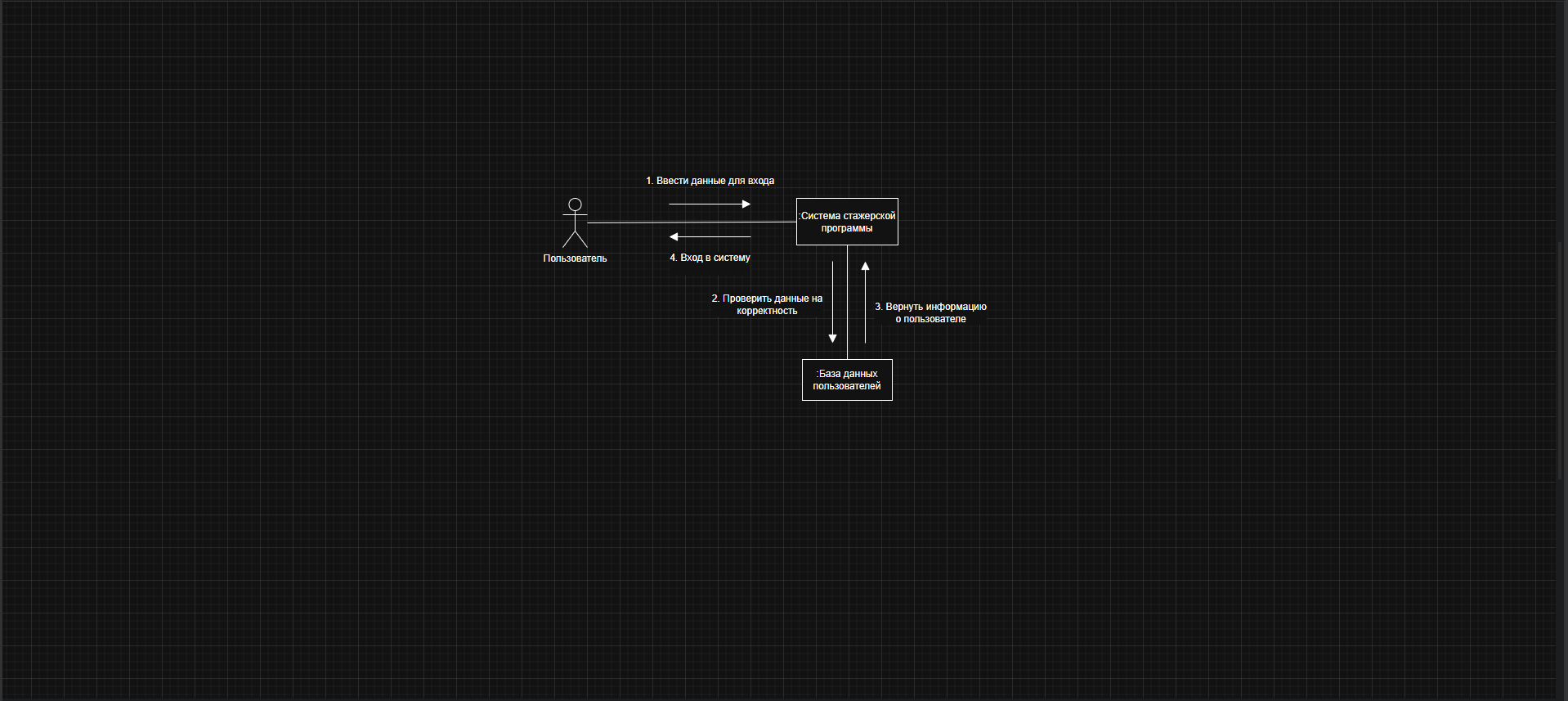
Основные элементы:

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

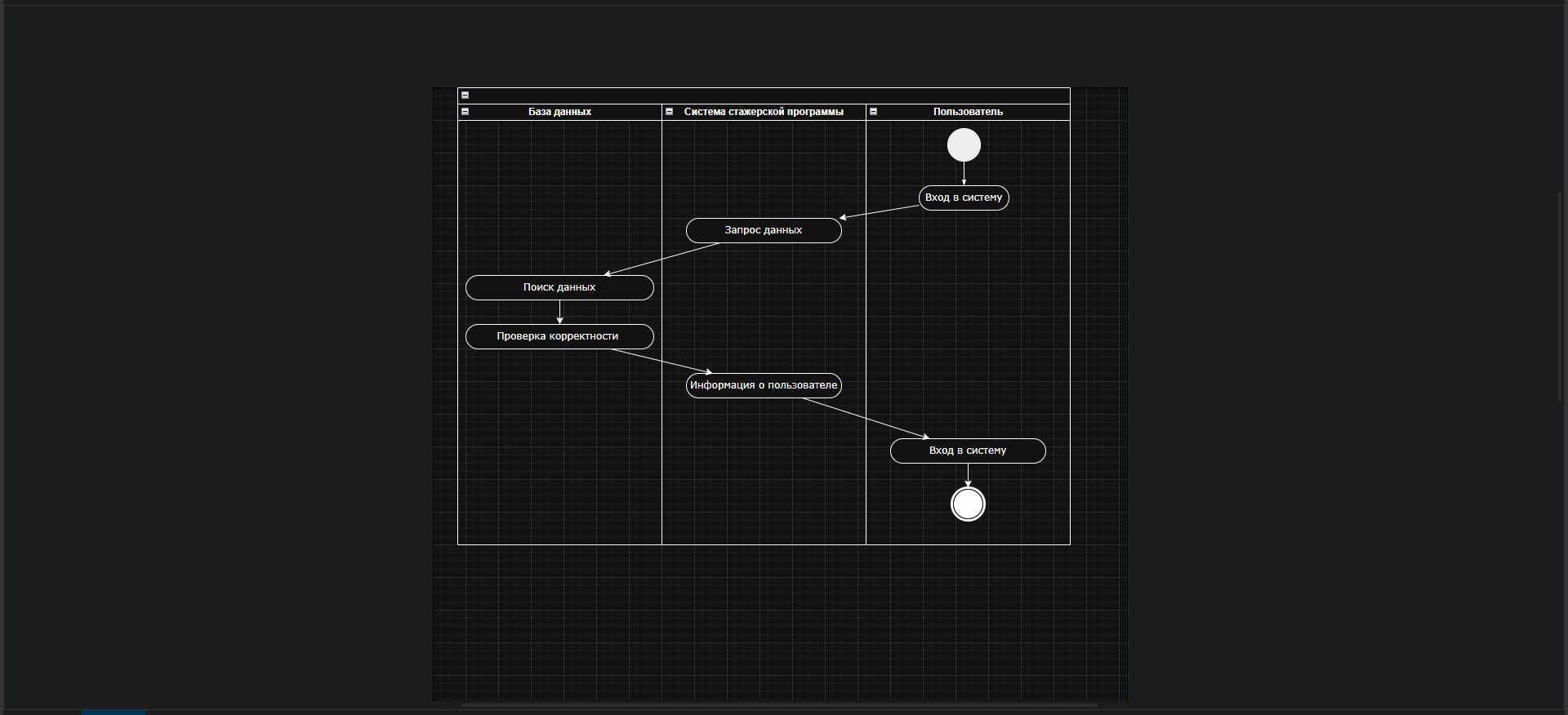
Для описания поведения и работы информационной системы Intern Guide, которая помогает стажерам лучше осваиваться на новом месте и получать различные советы, были разработаны следующие несколько диаграмм:

****

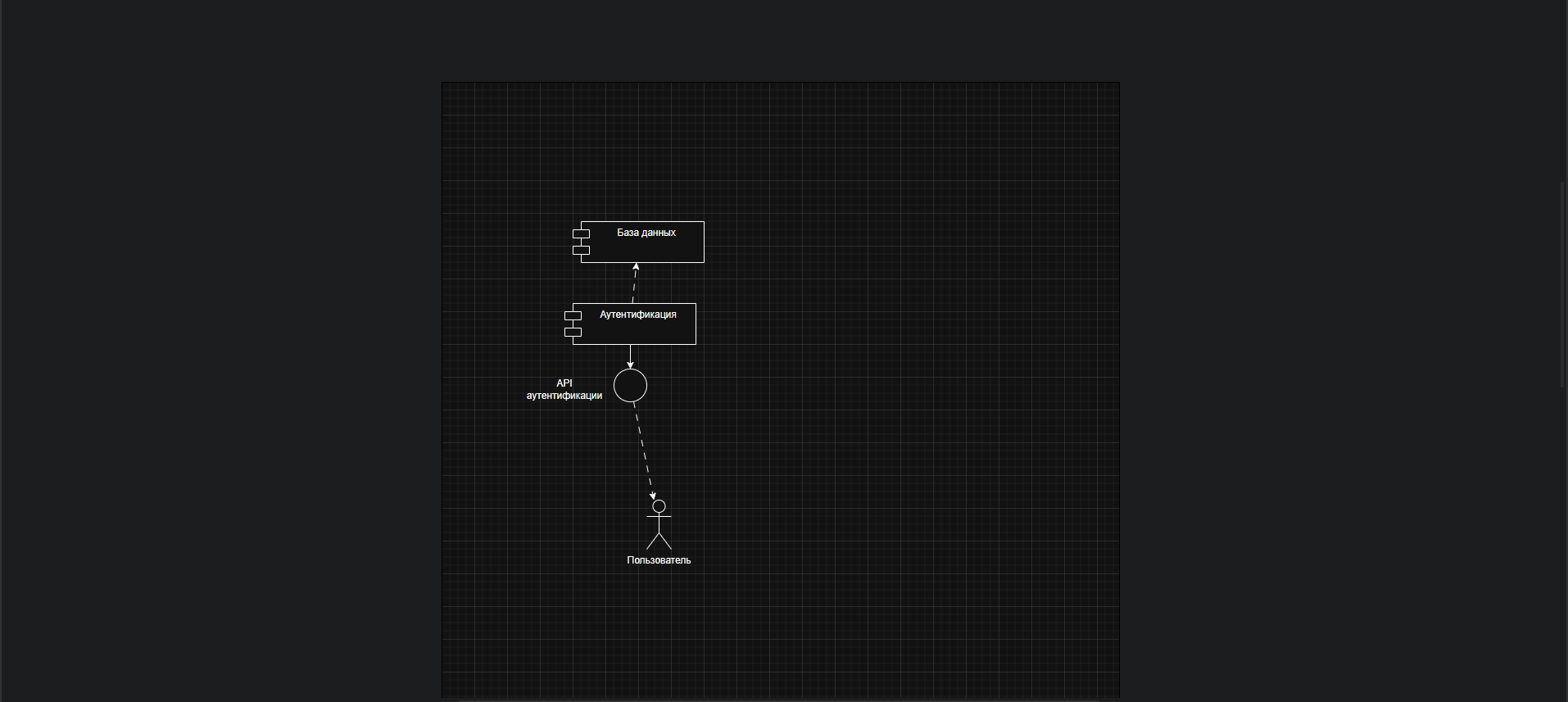
**Рисунок 1. Авторизация - диаграмма классов**

****

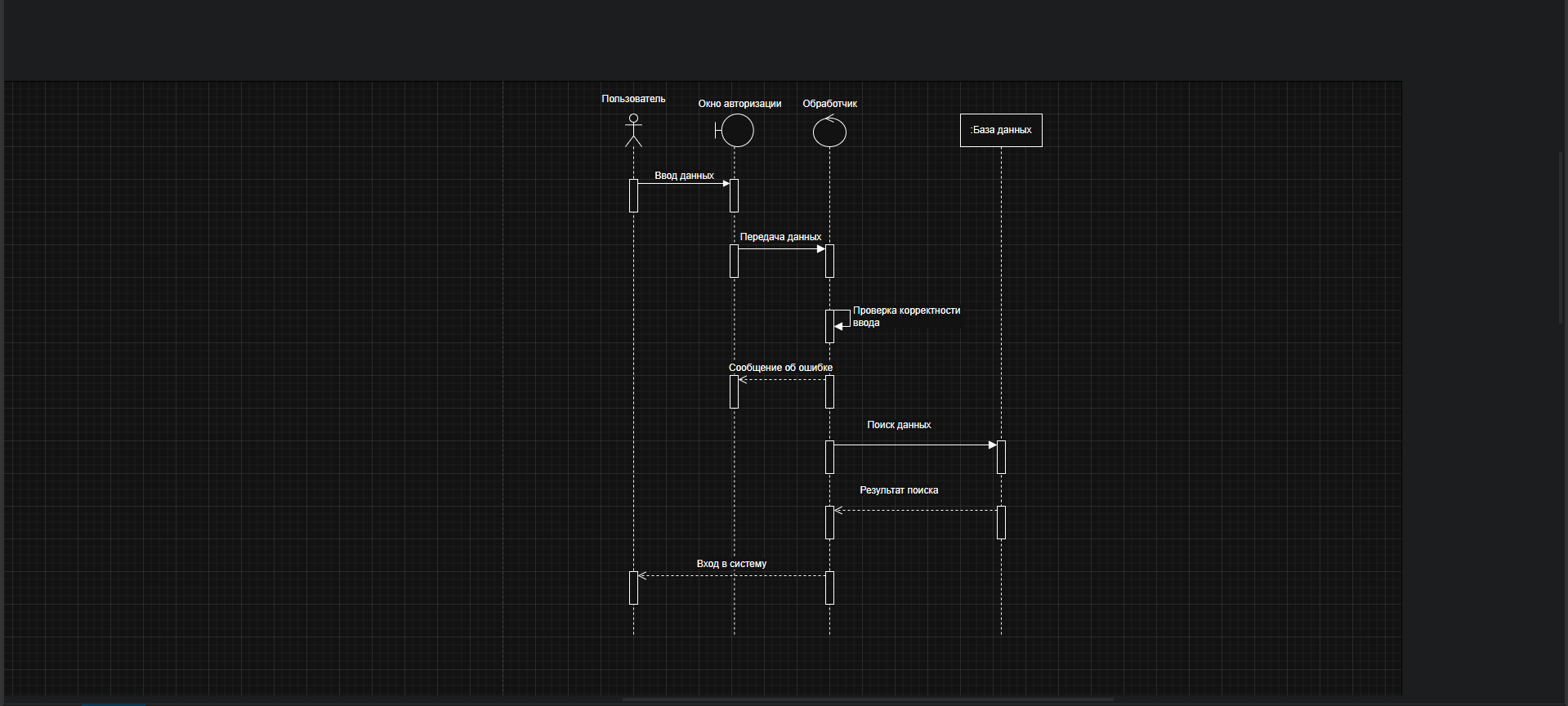
**Рисунок 2. Авторизация - диаграмма коопераций**

****

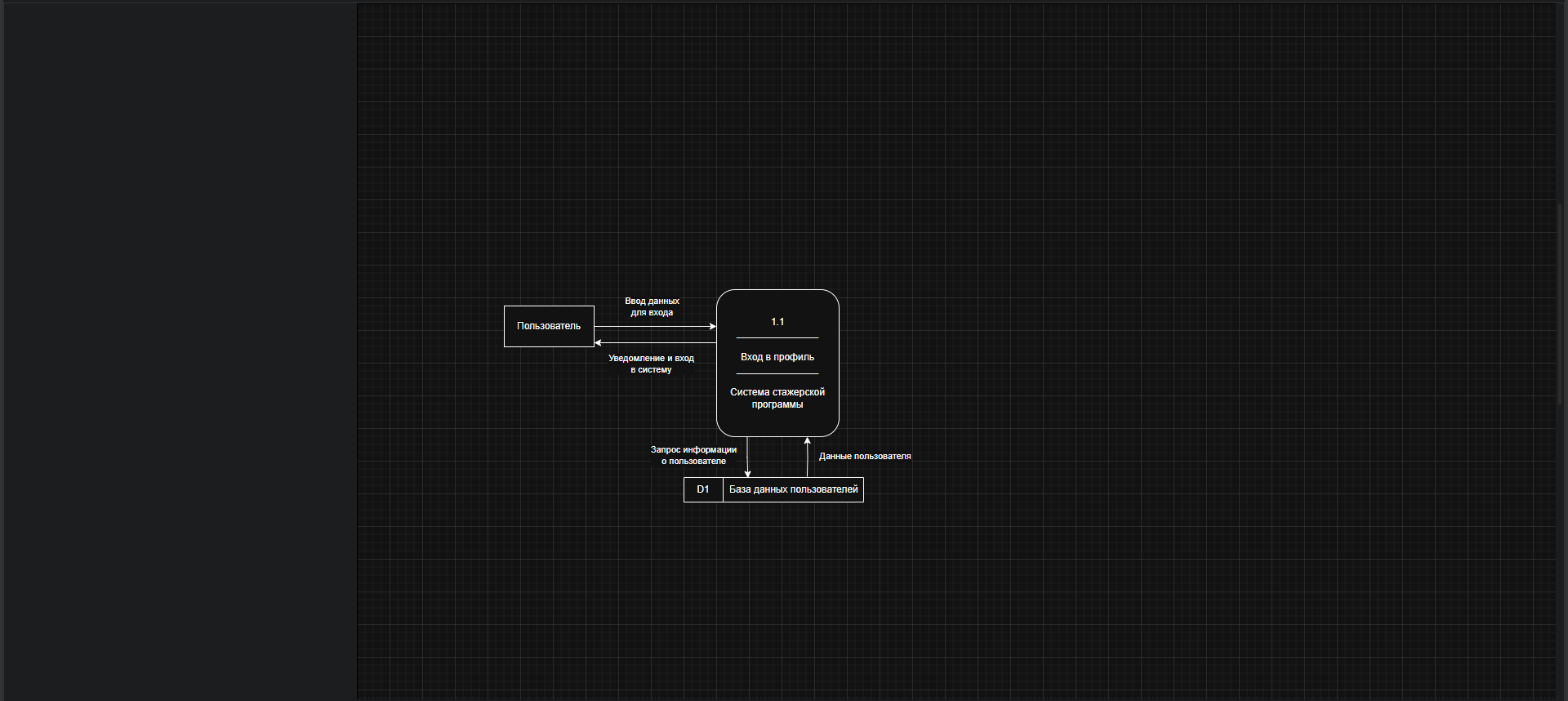
**Рисунок 3. Авторизация - диаграмма деятельности**

****

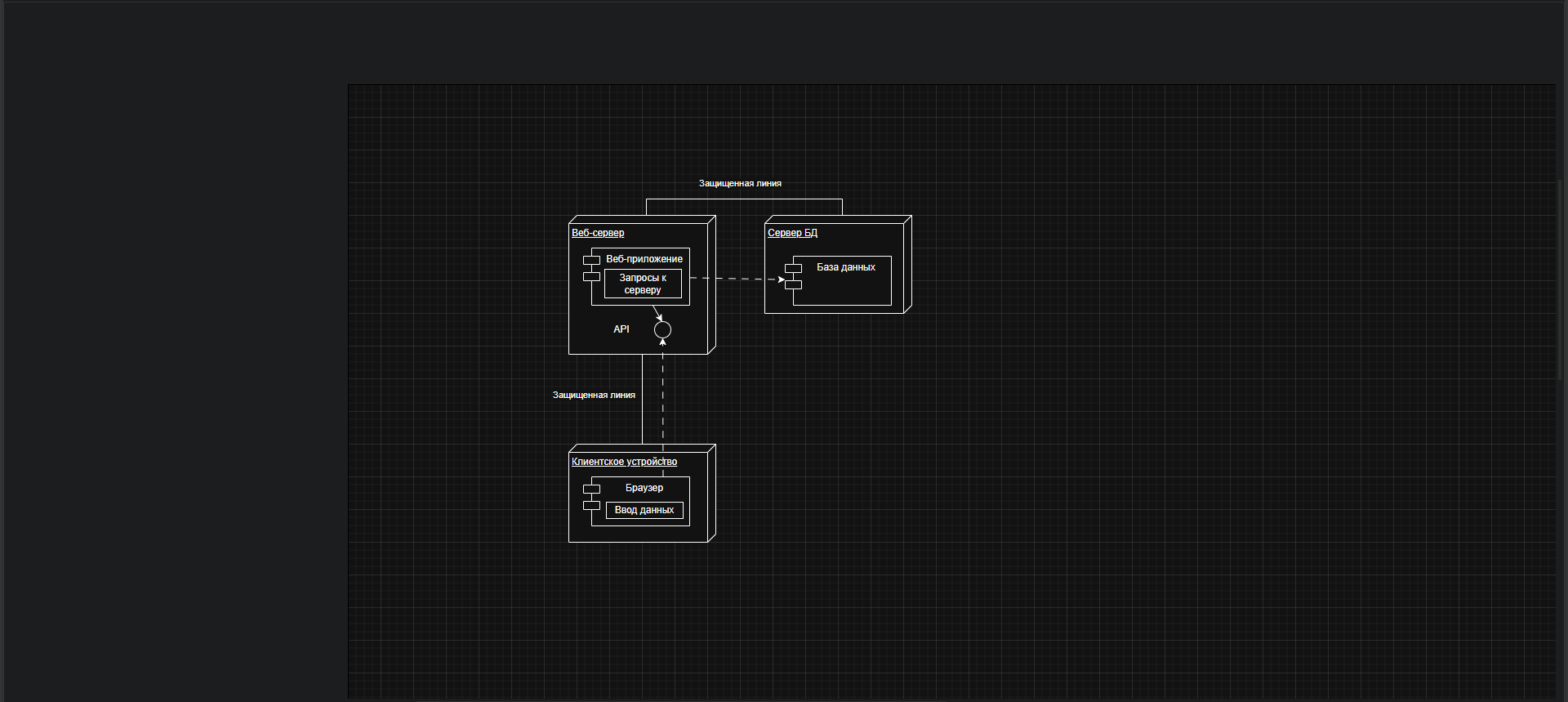
**Рисунок 4. Авторизация - диаграмма компонентов**

****

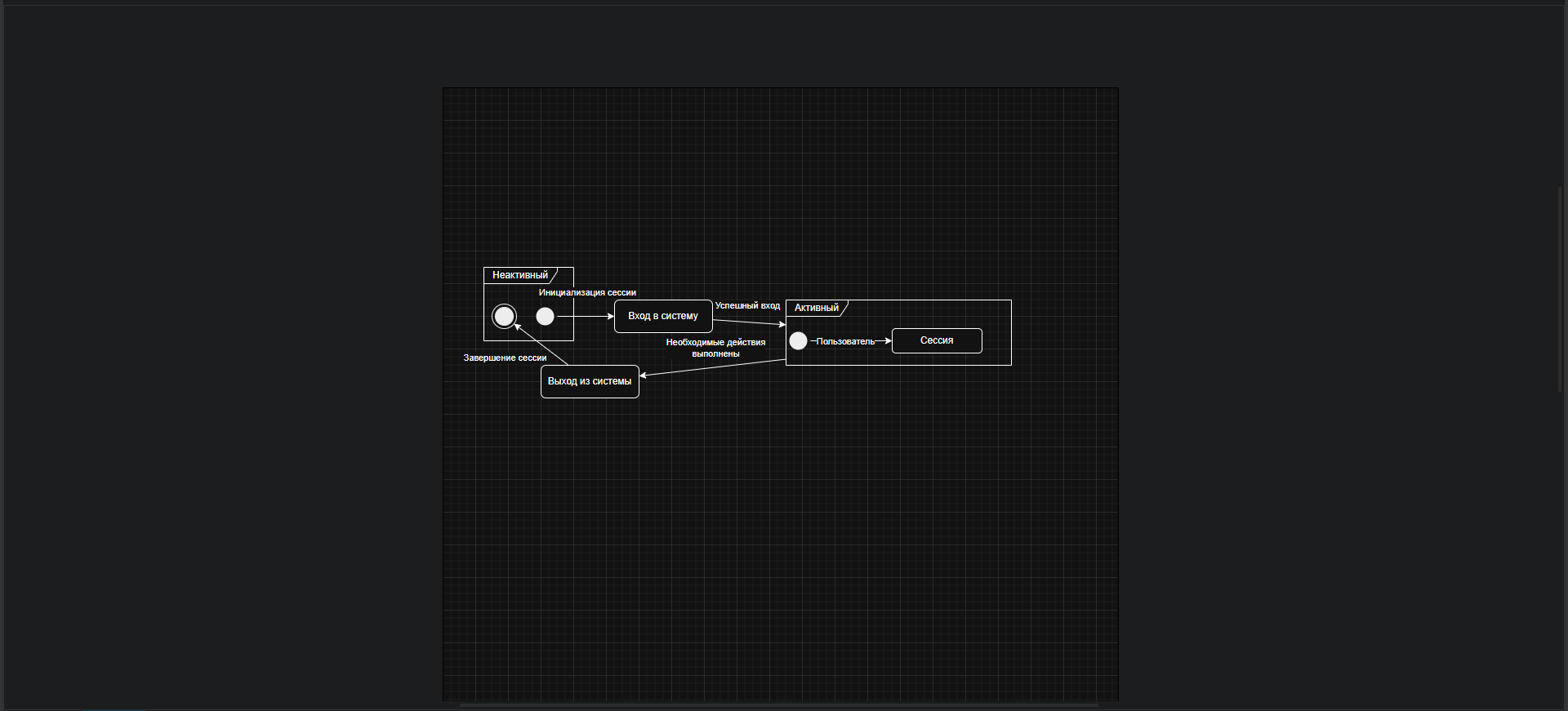
**Рисунок 5. Авторизация - диаграмма последовательности**

****

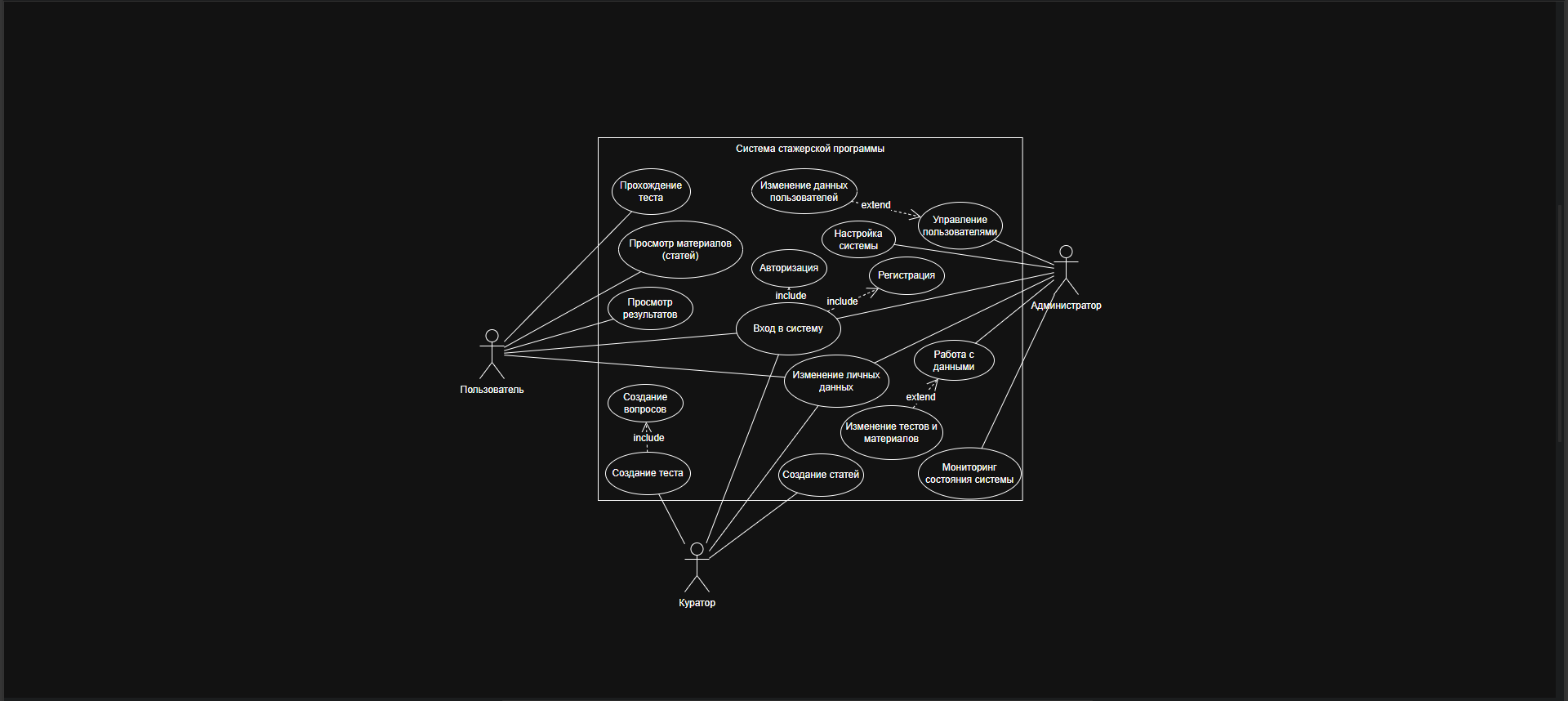
**Рисунок 6. Авторизация - диаграмма потоков данных**

****

**Рисунок 7. Авторизация - диаграмма развертывания**

****

**Рисунок 8. Авторизация - диаграмма состояния**

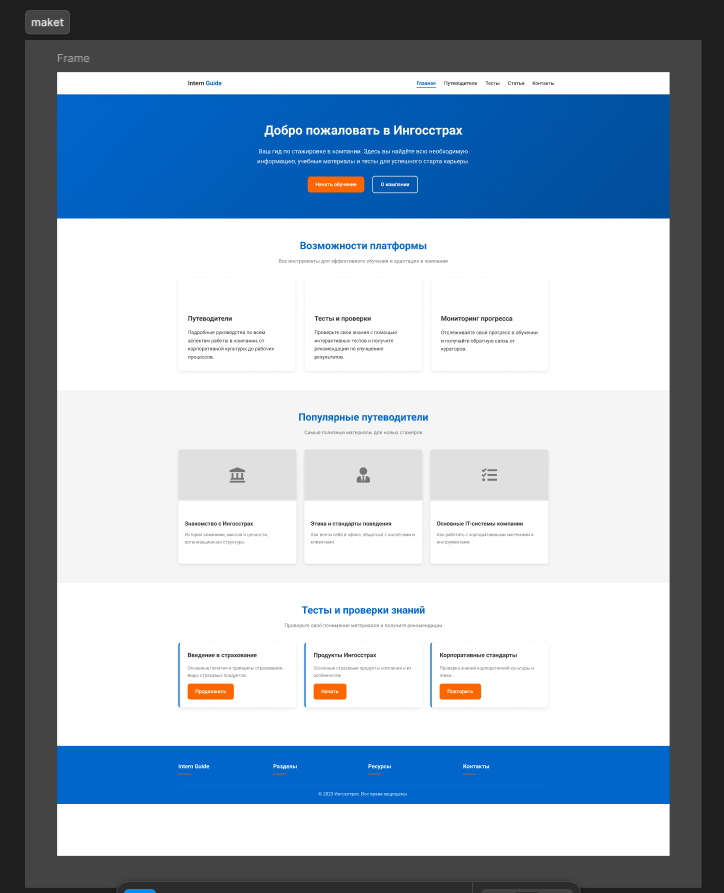
****

**Рисунок 9. 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.

Для визуального проектирования пользовательского интерфейса сайта был разработан макет. Данный макет содержит основную информацию, которая будет представлена пользователю и будет служить опорой для реализации продукта.

****

**Рисунок 10. Макет сайта Intern Guide**

## **2.4 Разработка программного модуля**

Intern Guide

## **2.5 Тестирование программного продукта**

Разработка тестовой документации для системы адаптации стажёров Intern Guide проводилась с особой тщательностью, учитывая важность системы для бизнес-процессов компании. Качество тестирования напрямую влияет на эффективность адаптации новых сотрудников.

1. **Анализ требований и подготовка**

На начальном этапе были выполнены следующие действия:

* Проведен детальный анализ технического задания с выделением ключевых требований
* Определены приоритетные направления тестирования
* Составлен перечень возможных рисков

Были выбраны основные виды тестирования:

* Функциональное тестирование всех модулей системы
* Интеграционная проверка взаимодействия компонентов
* Юзабилити-тестирование интерфейса

1. **Подготовка тестовых данных**

Для обеспечения комплексной проверки системы были подготовлены:

* Детализированные тест-кейсы с пошаговыми инструкциями
* Чек-листы для проверки ключевых функций
* Сценарии проверки граничных условий

Каждый тест-кейс содержал:

* Четкую последовательность действий
* Ожидаемые результаты
* Критерии успешного прохождения

1. **Планирование и проведение тестирования**

Был разработан детальный план тестирования. После каждого этапа тестирования формировались отчеты, содержащие:

* Общее количество выполненных тестов
* Найденные дефекты с указанием приоритетов

Реализованный подход к тестированию позволил:

* Выявить и устранить критические ошибки до запуска системы
* Оптимизировать пользовательский интерфейс
* Обеспечить стабильную работу под нагрузкой
* Повысить общее качество продукта

Таким образом, подход к тестированию PlantDoctor был максимально детализированным, что позволило создать надежный и удобный продукт для пользователей.

Для проведения тестирования функциональных возможностей программного продукта, а также взаимодействия модулей и компонентов между собой были разработаны модули автоматизированного тестирования (UNIT-тесты), которые позволяют удостовериться в правильности прохождения функциями тестов, а также ускоряют сам процесс тестирования.

## **2.6 Разработка модуля математического моделирования**

ага

# **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 Flask, что обеспечило высокую производительность API и простоту интеграции с существующей инфраструктурой компании. Frontend-часть разрабатывалась при помощи разметки HTML и реализации функций на JavaScript, 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.

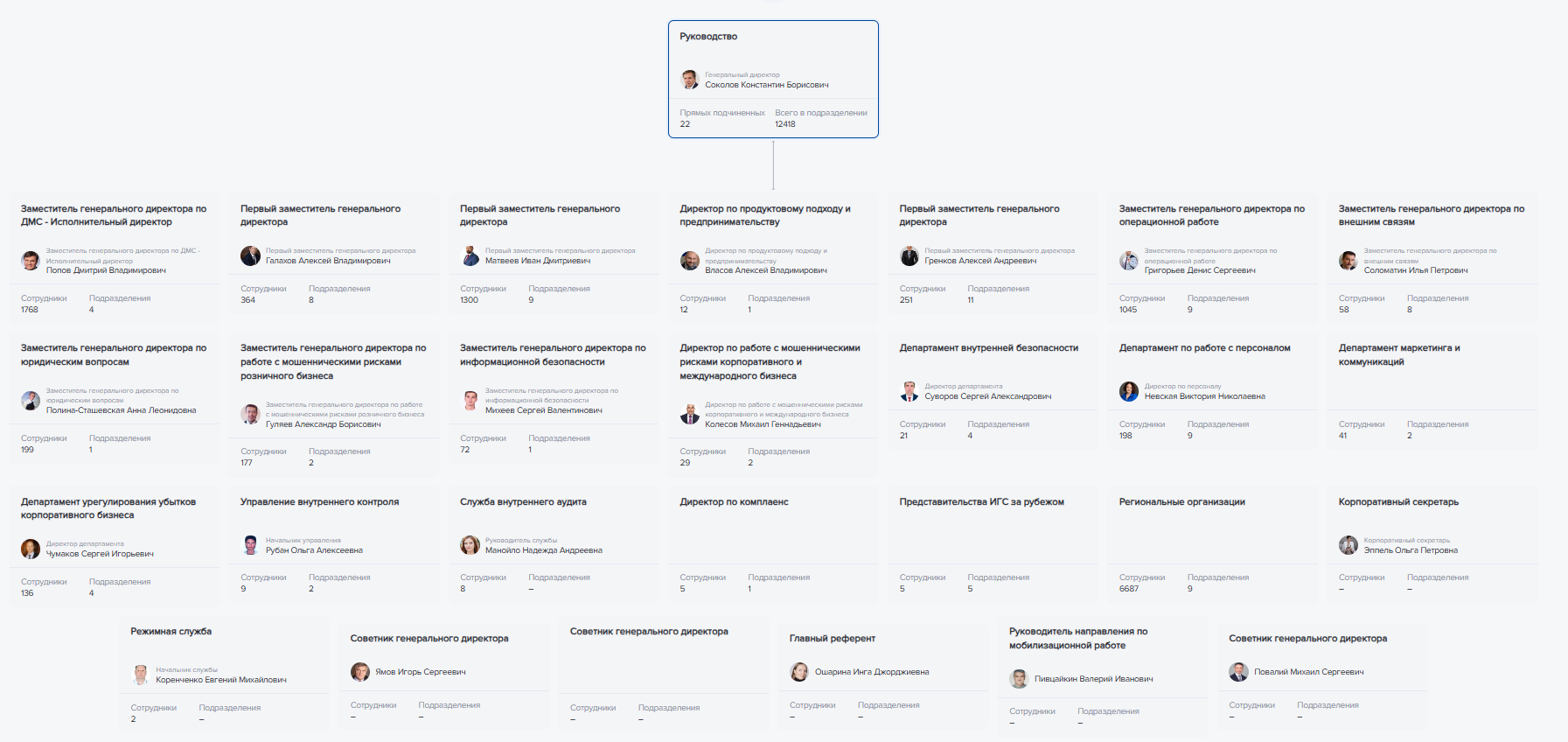
# **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

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.

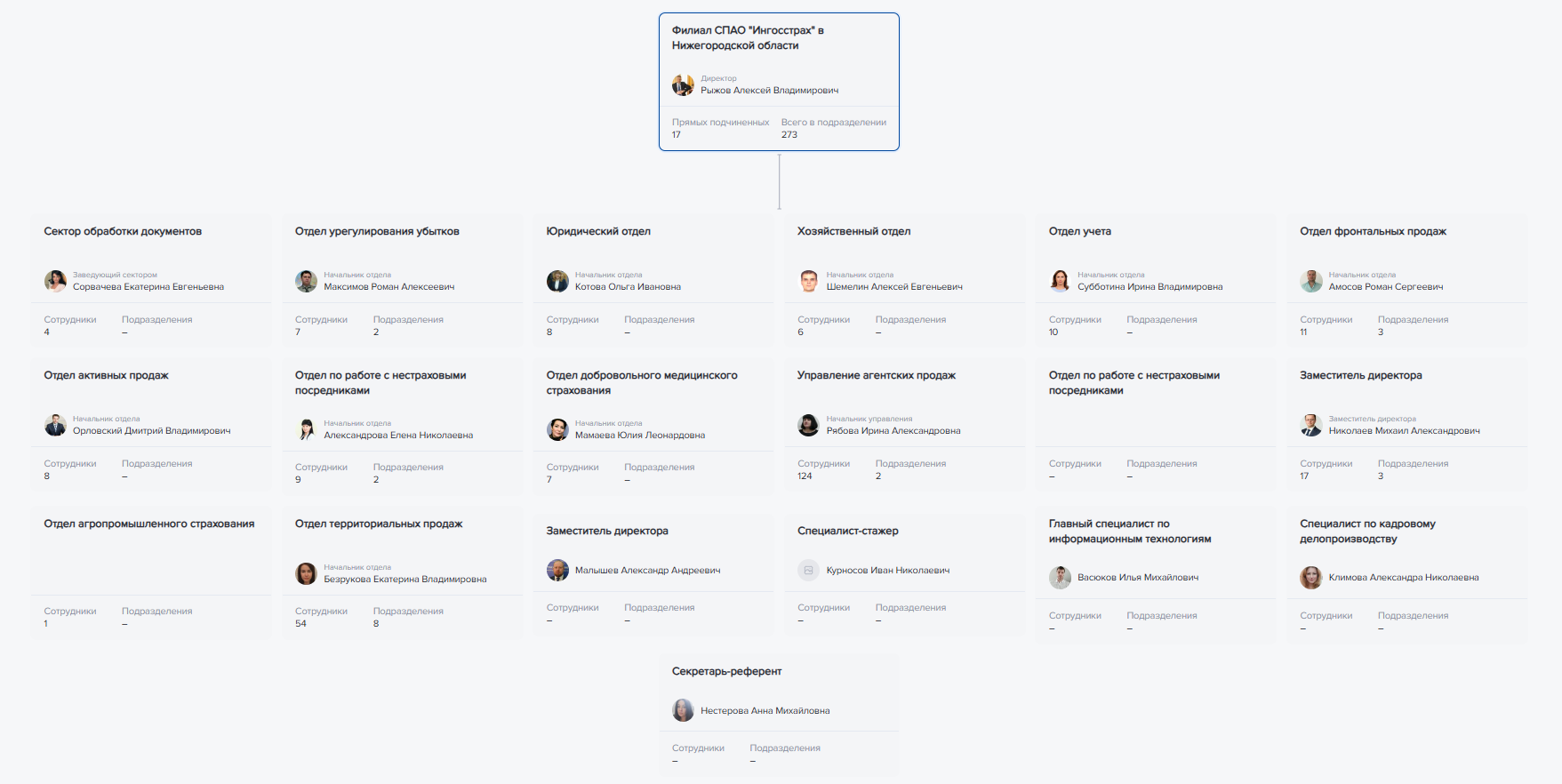
# **ПРИЛОЖЕНИЯ**

## **ПРИЛОЖЕНИЕ А**

Оргструктура предприятия

****

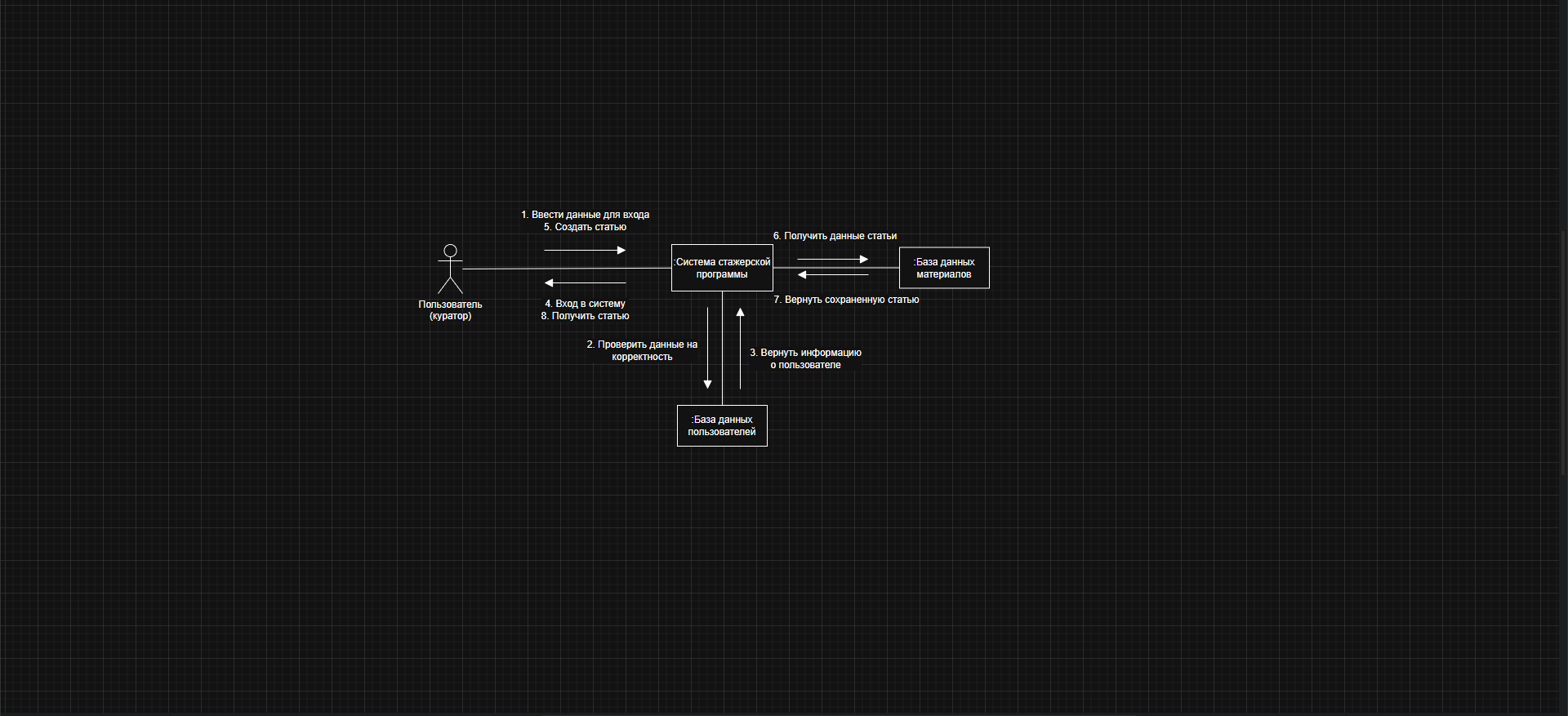
**Рисунок 11. Структура предприятия**

****

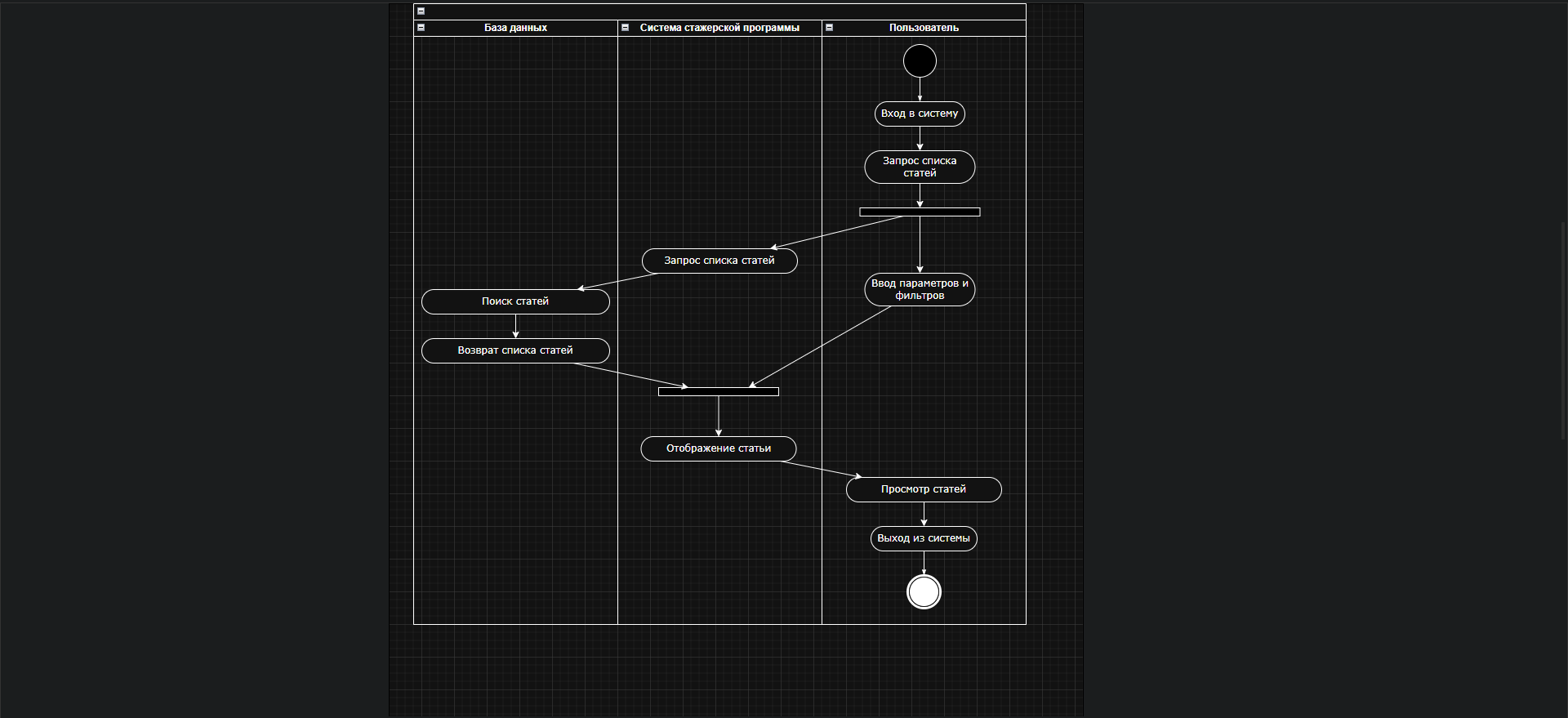
**Рисунок 12. Структура регионального предприятия**

## **ПРИЛОЖЕНИЕ Б**

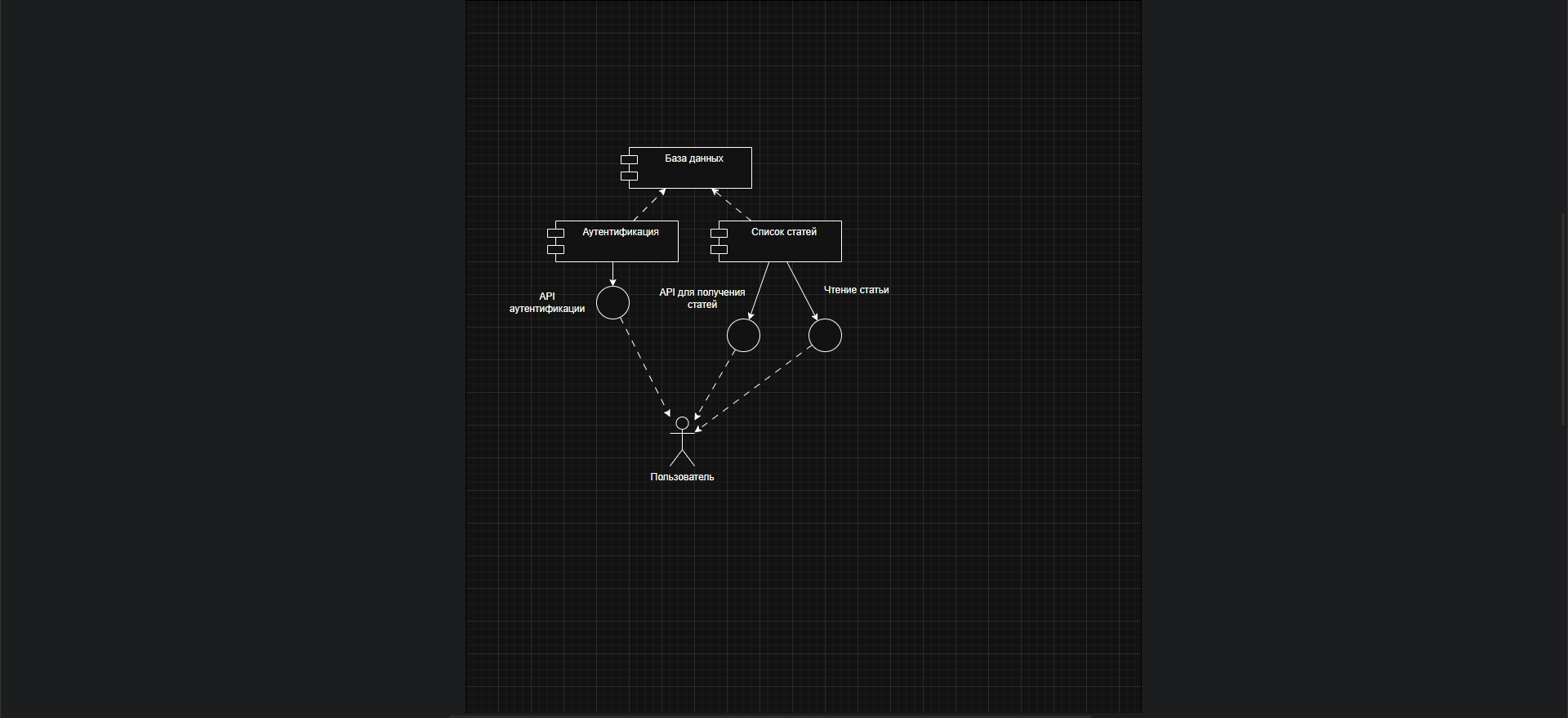
UML-диаграммы модулей программы

****

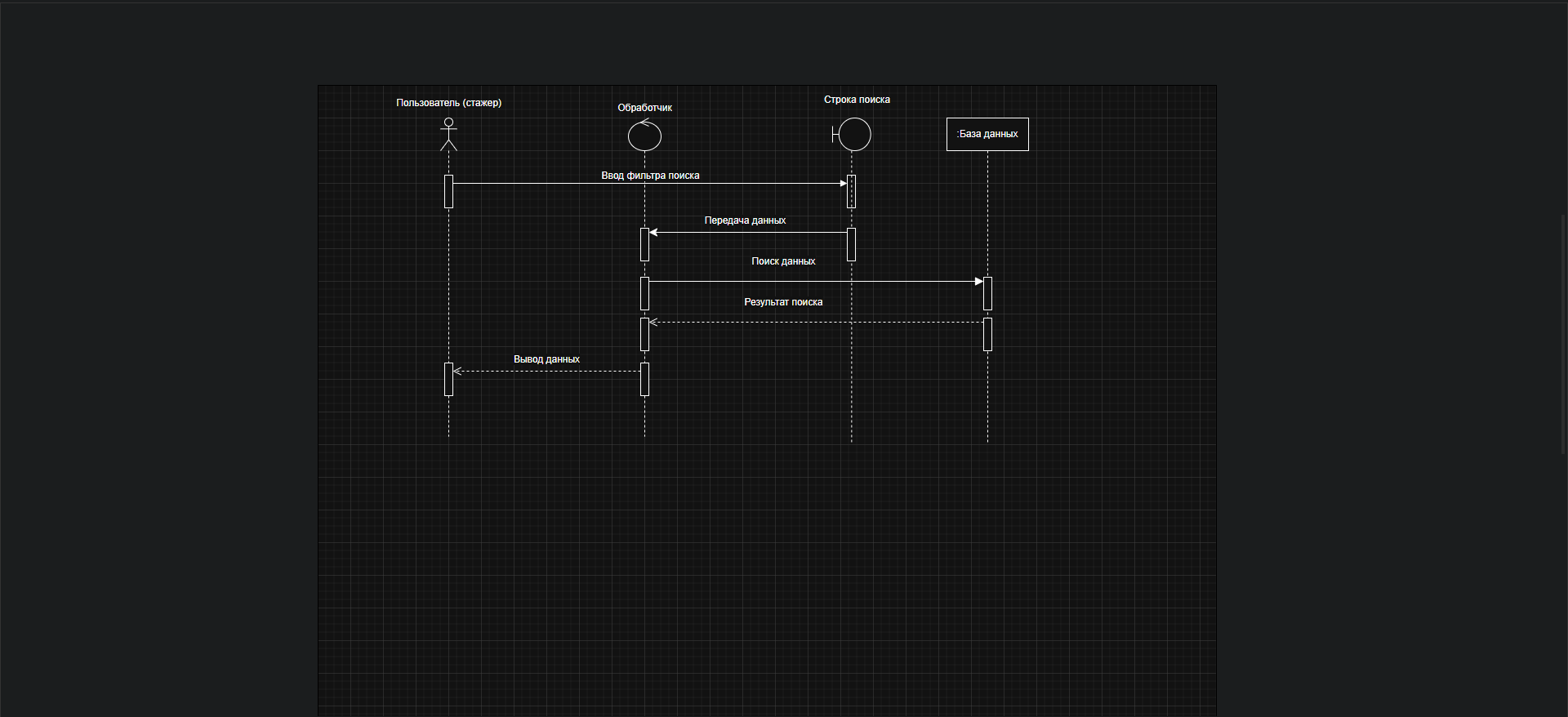
**Рисунок 13. Материалы - диаграмма коопераций**

****

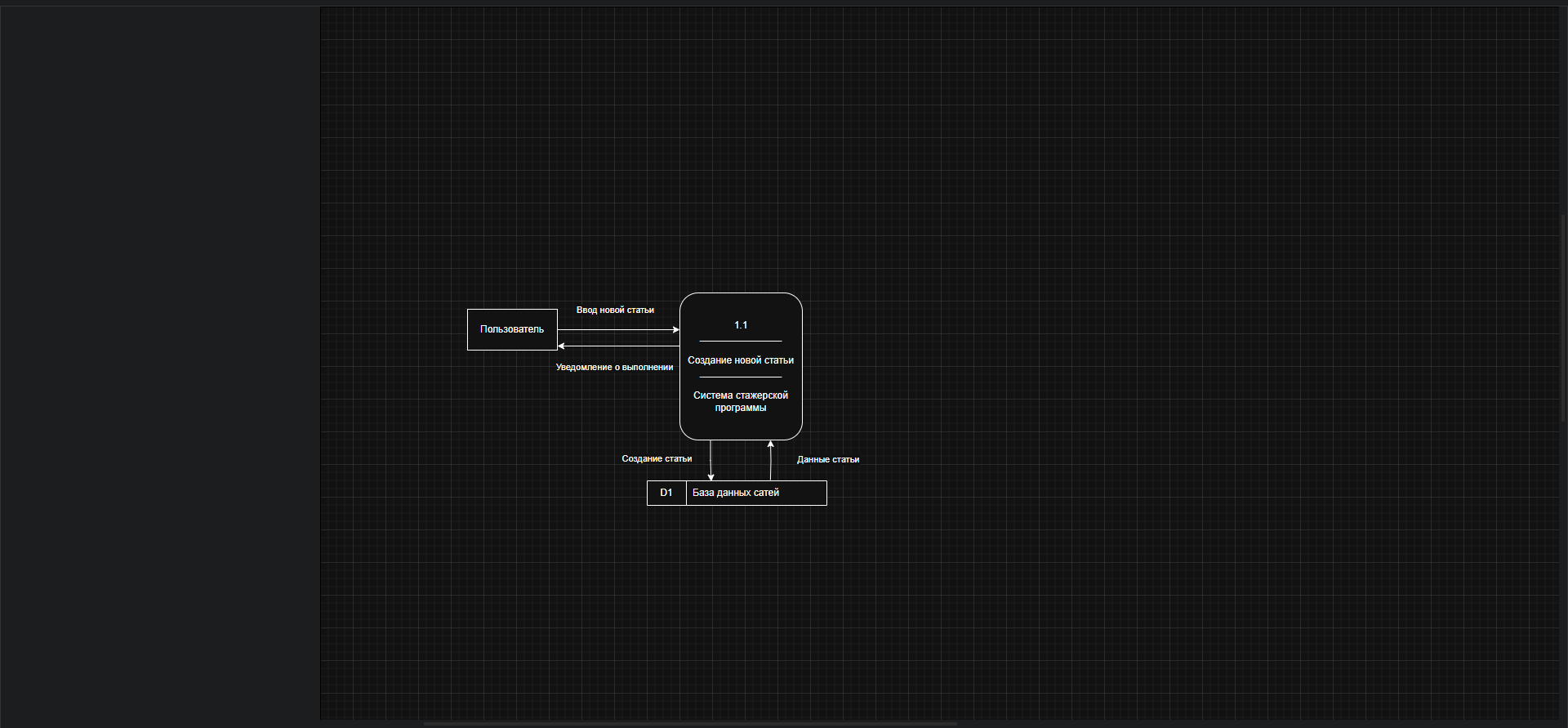
**Рисунок 14. Материалы - диаграмма деятельности**

****

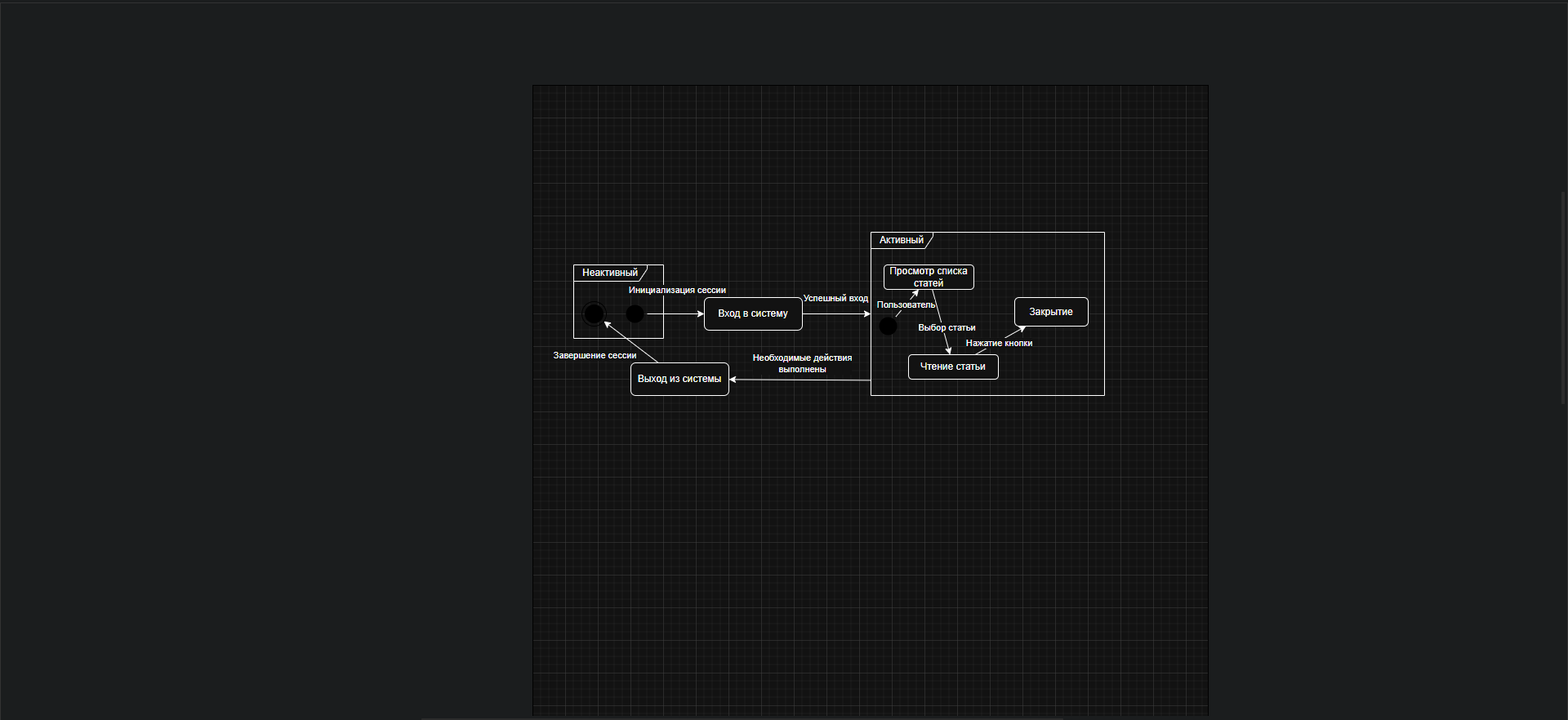
**Рисунок 15. Материалы - диаграмма компонентов**

****

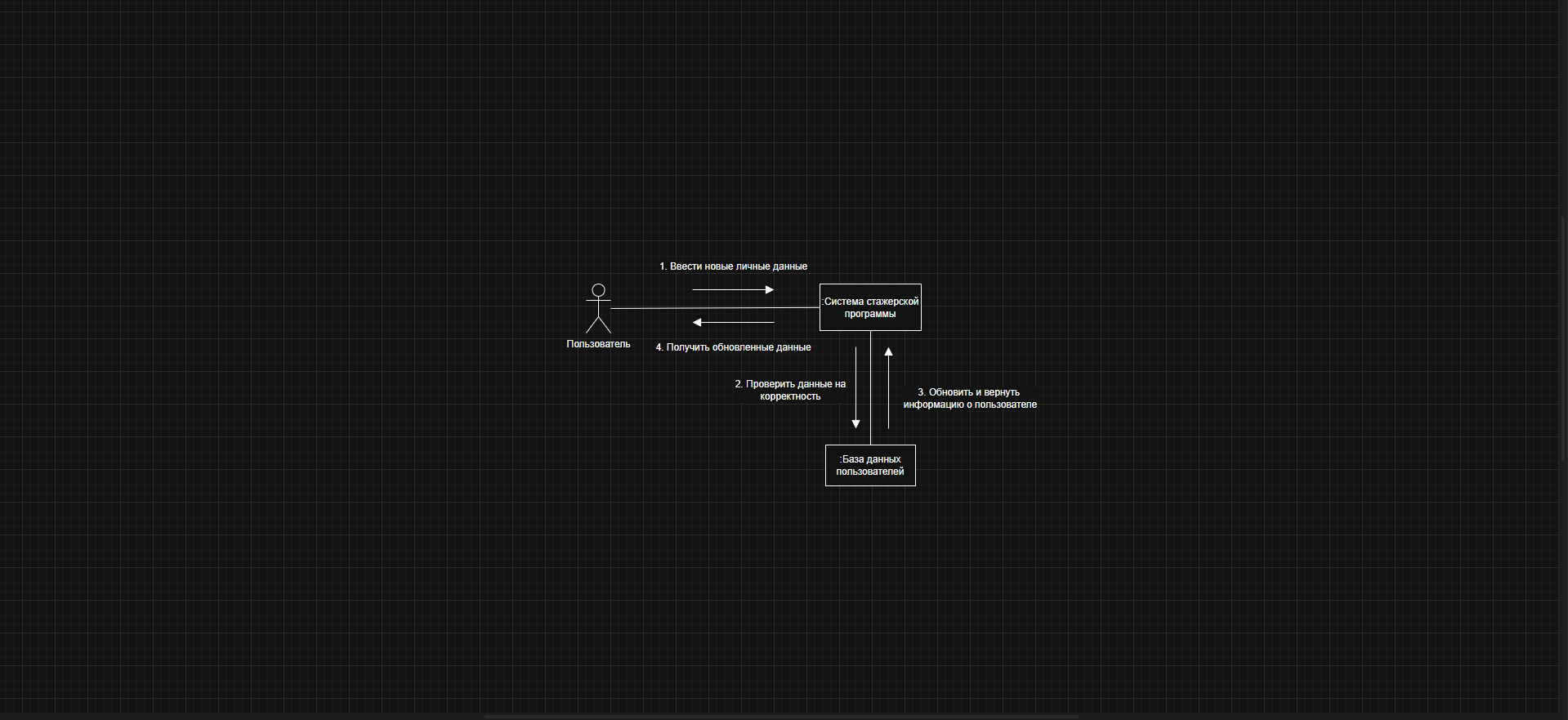
**Рисунок 16. Материалы - диаграмма последовательности**

****

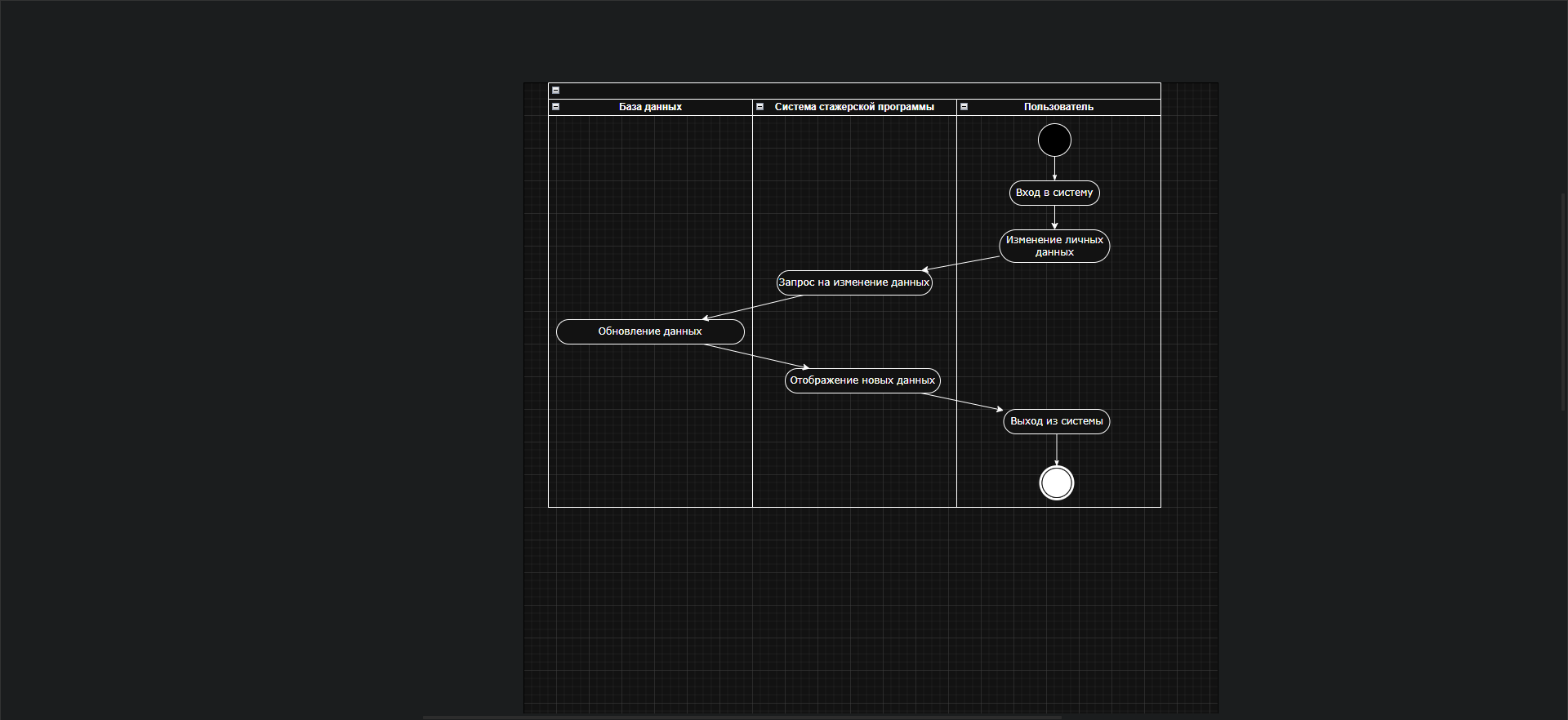
**Рисунок 17. Материалы - диаграмма потоков данных**

****

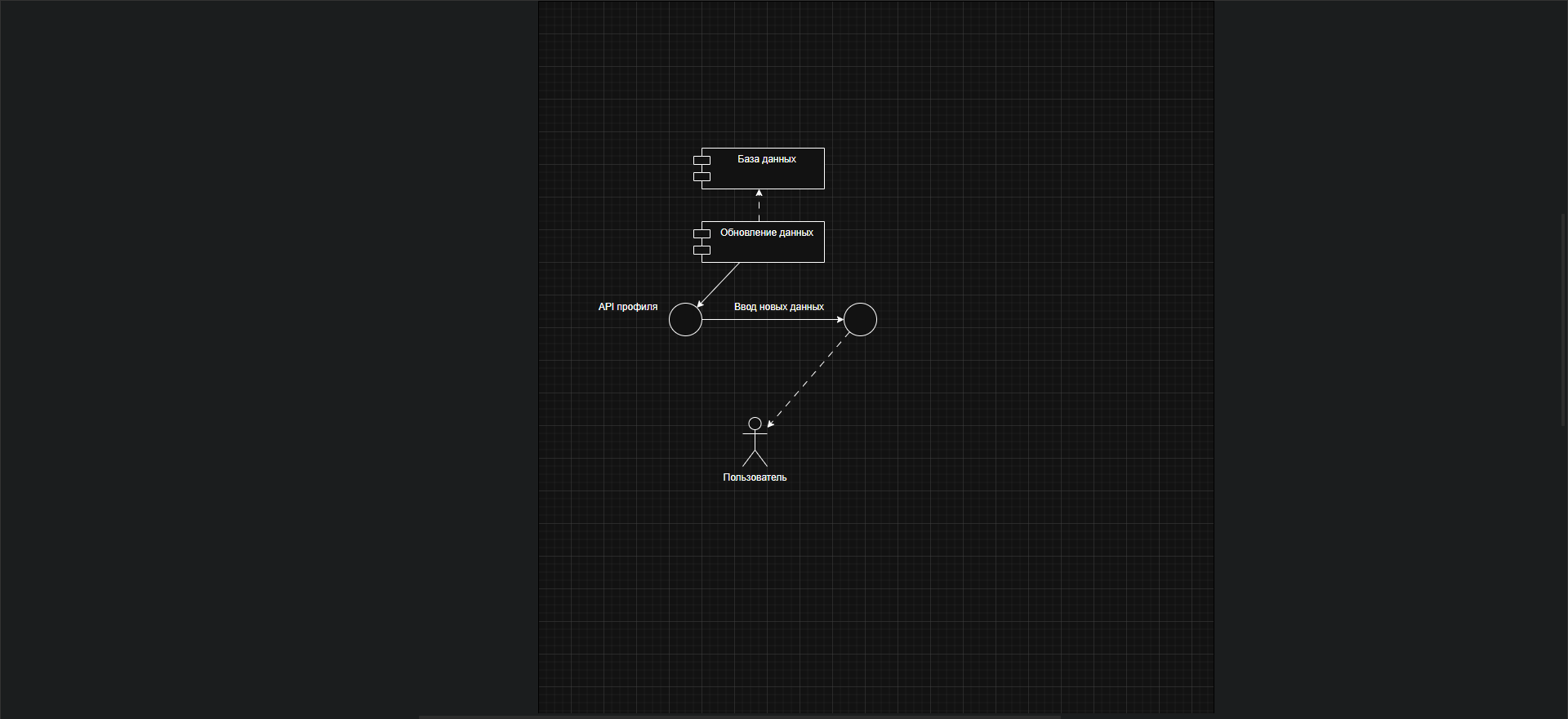
**Рисунок 18. Материалы - диаграмма состояния**

****

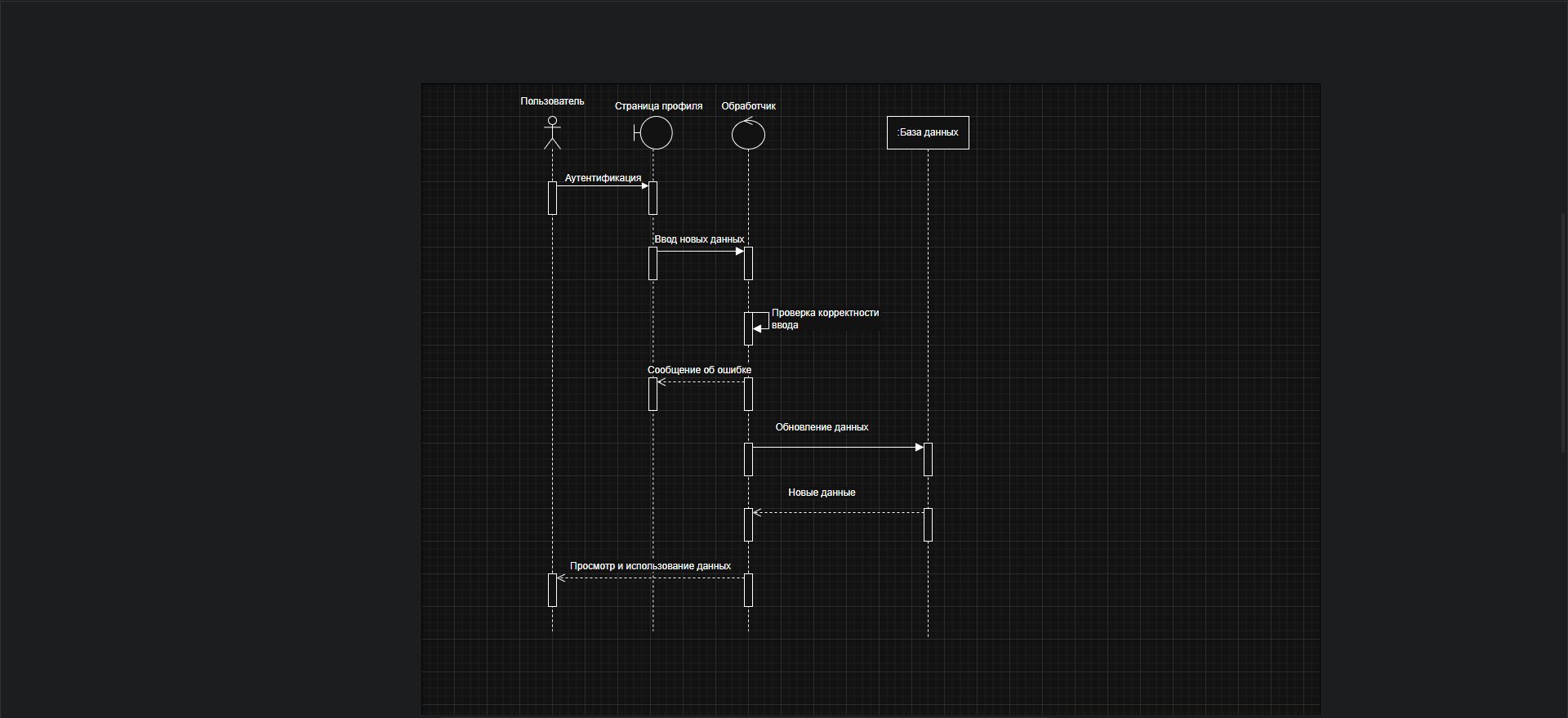
**Рисунок 19. Обновление данных - диаграмма кооперации**

****

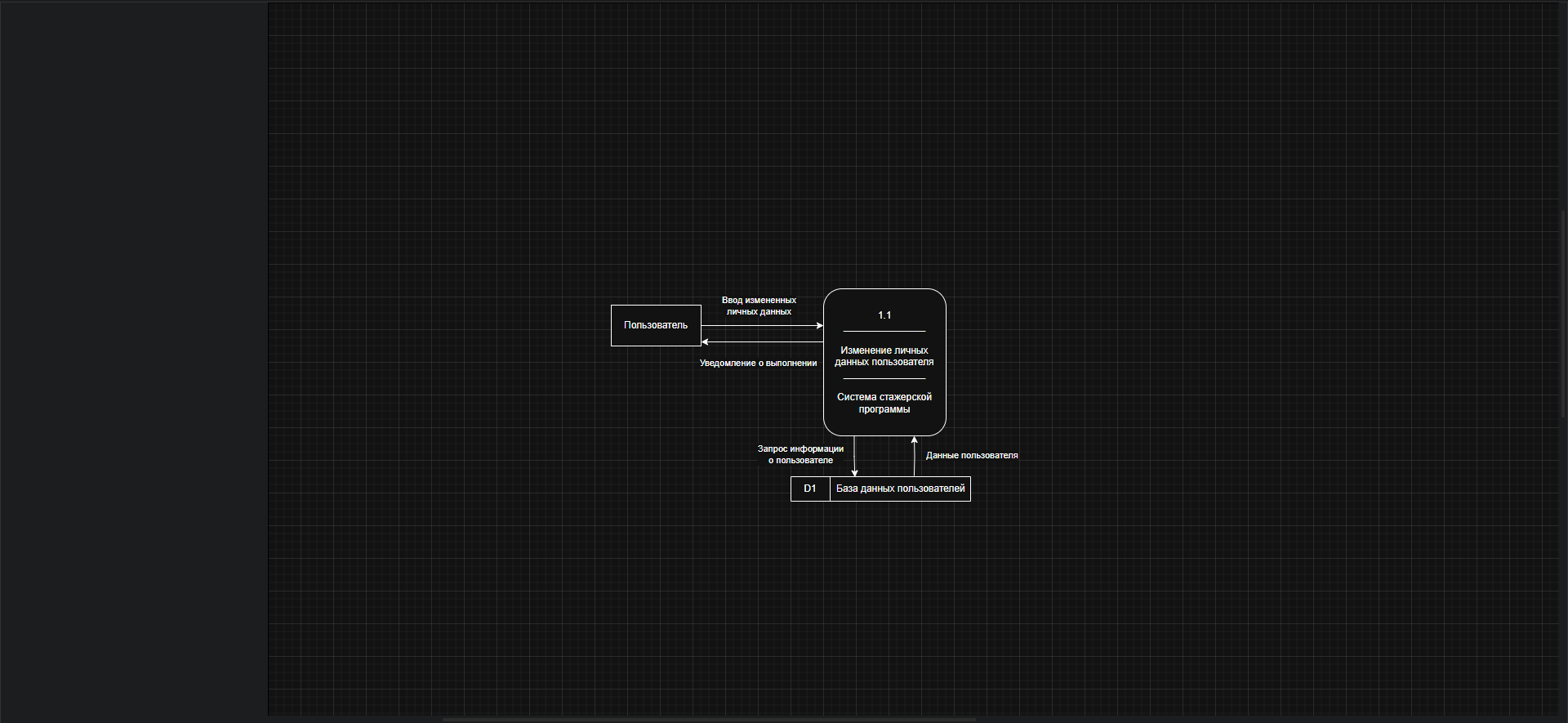
**Рисунок 20. Обновление данных - диаграмма деятельности**

****

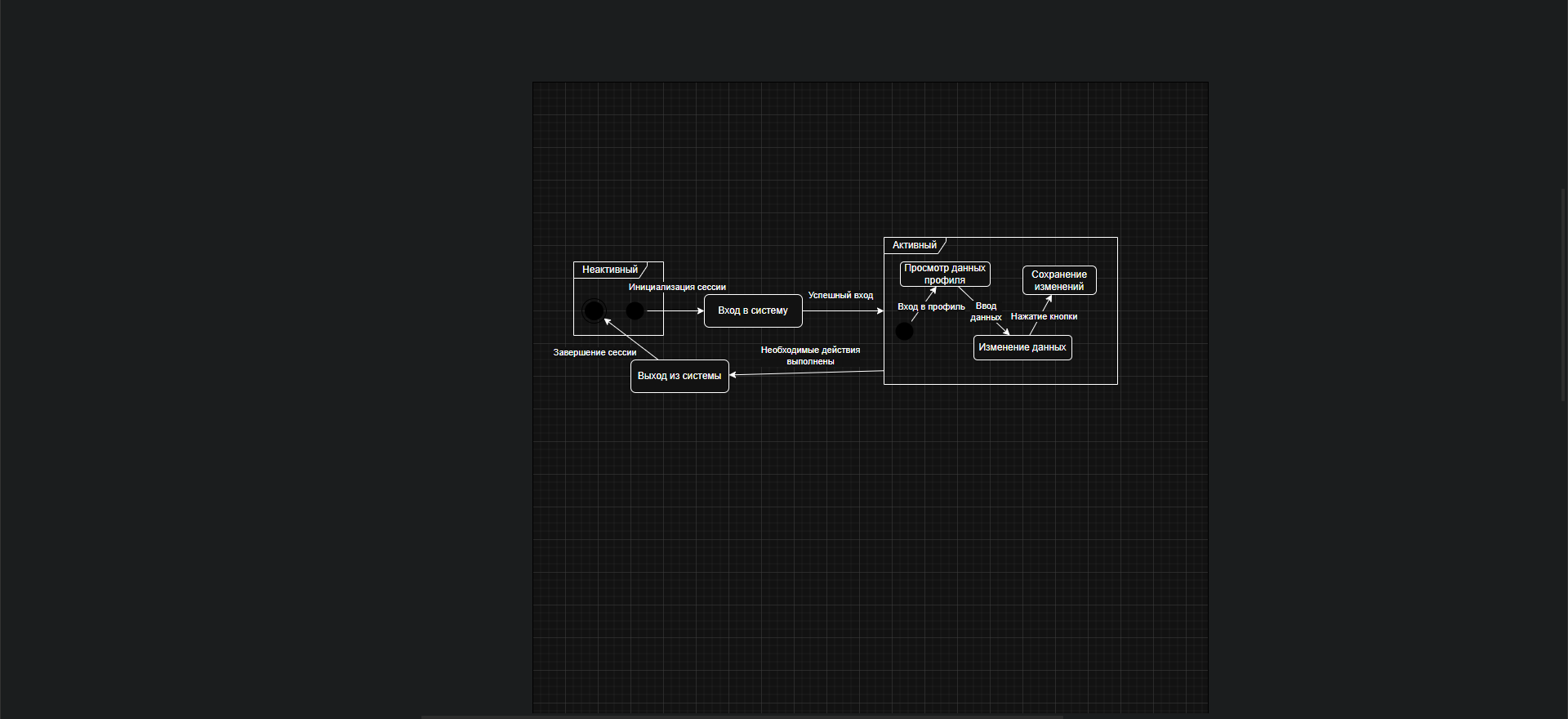
**Рисунок 21. Обновление данных - диаграмма компонентов**

****

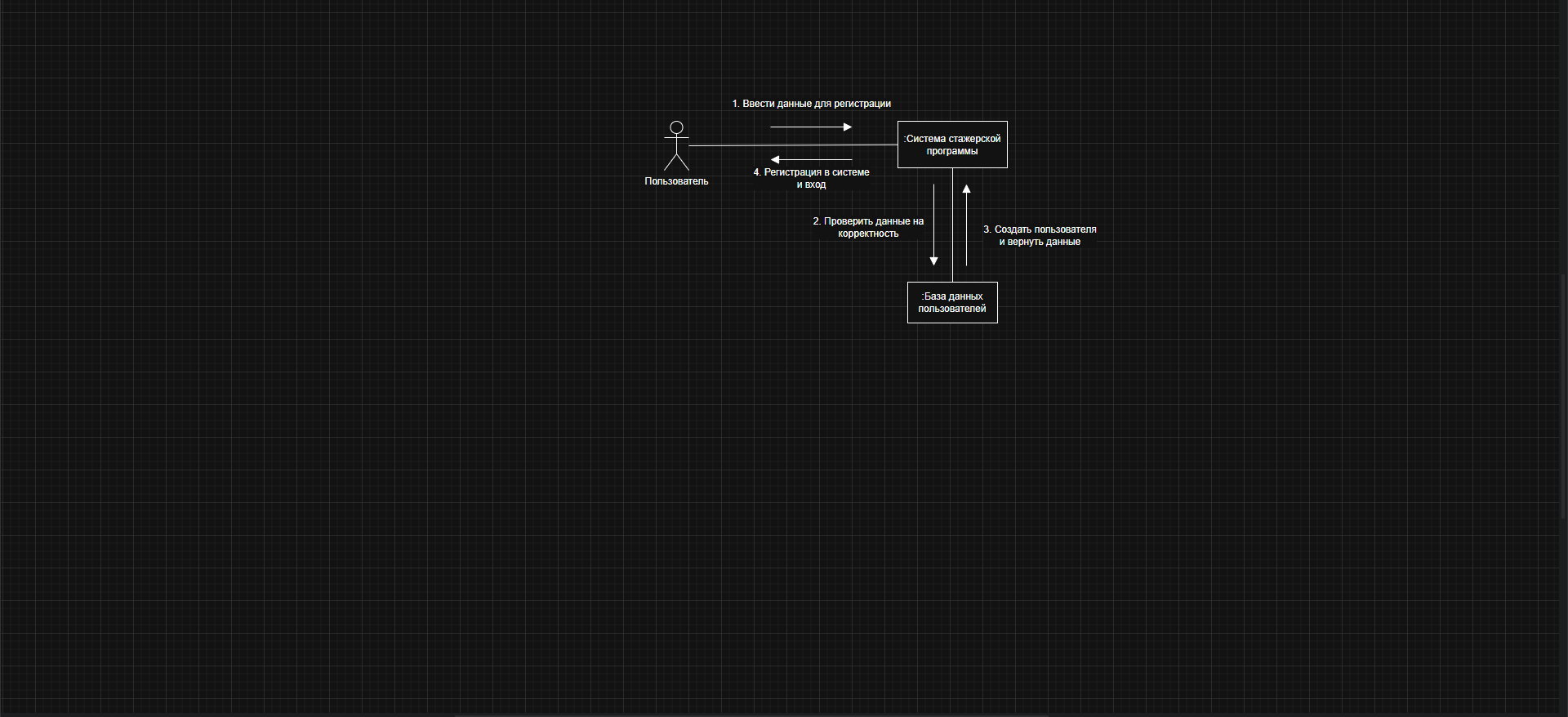
**Рисунок 22. Обновление данных - диаграмма последовательности**

****

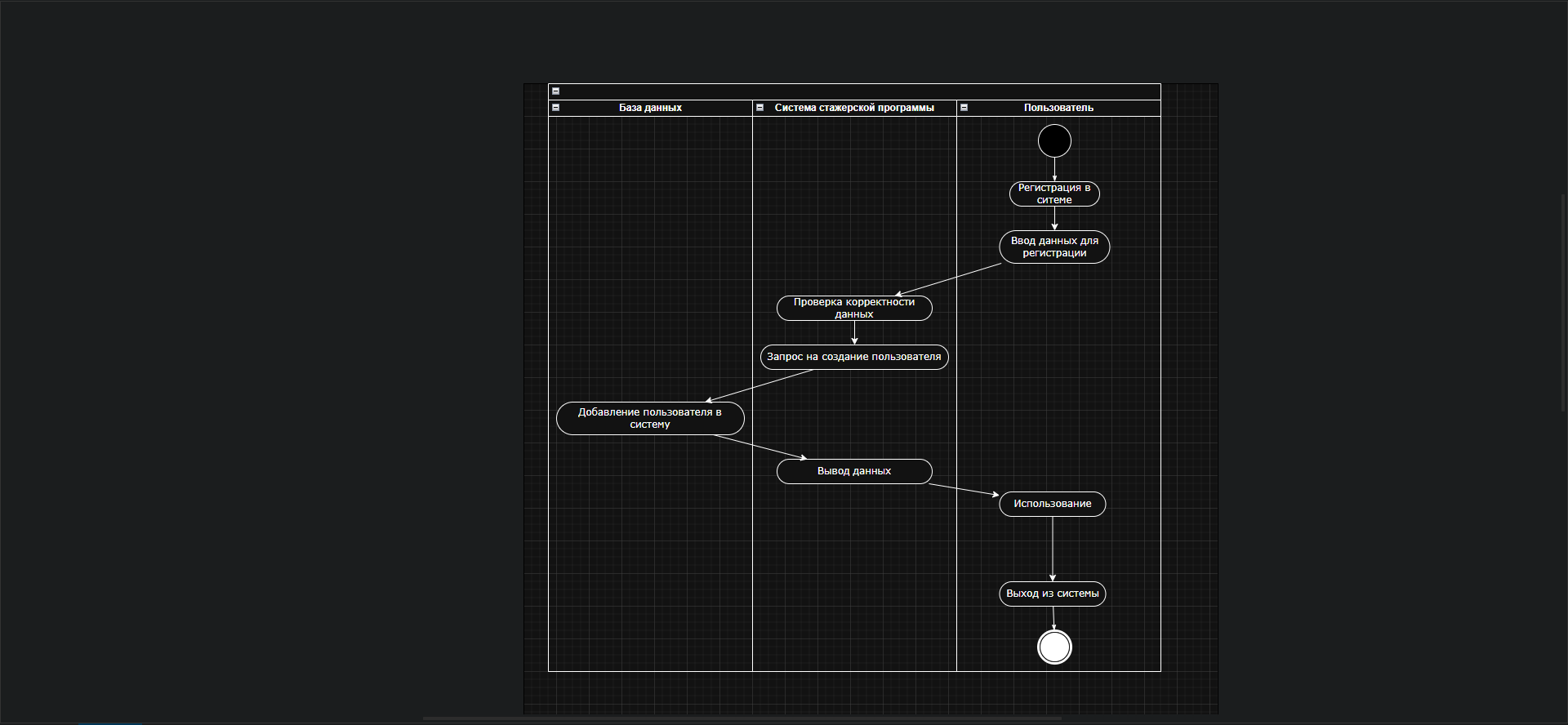
**Рисунок 23. Обновление данных - диаграмма потоков данных**

****

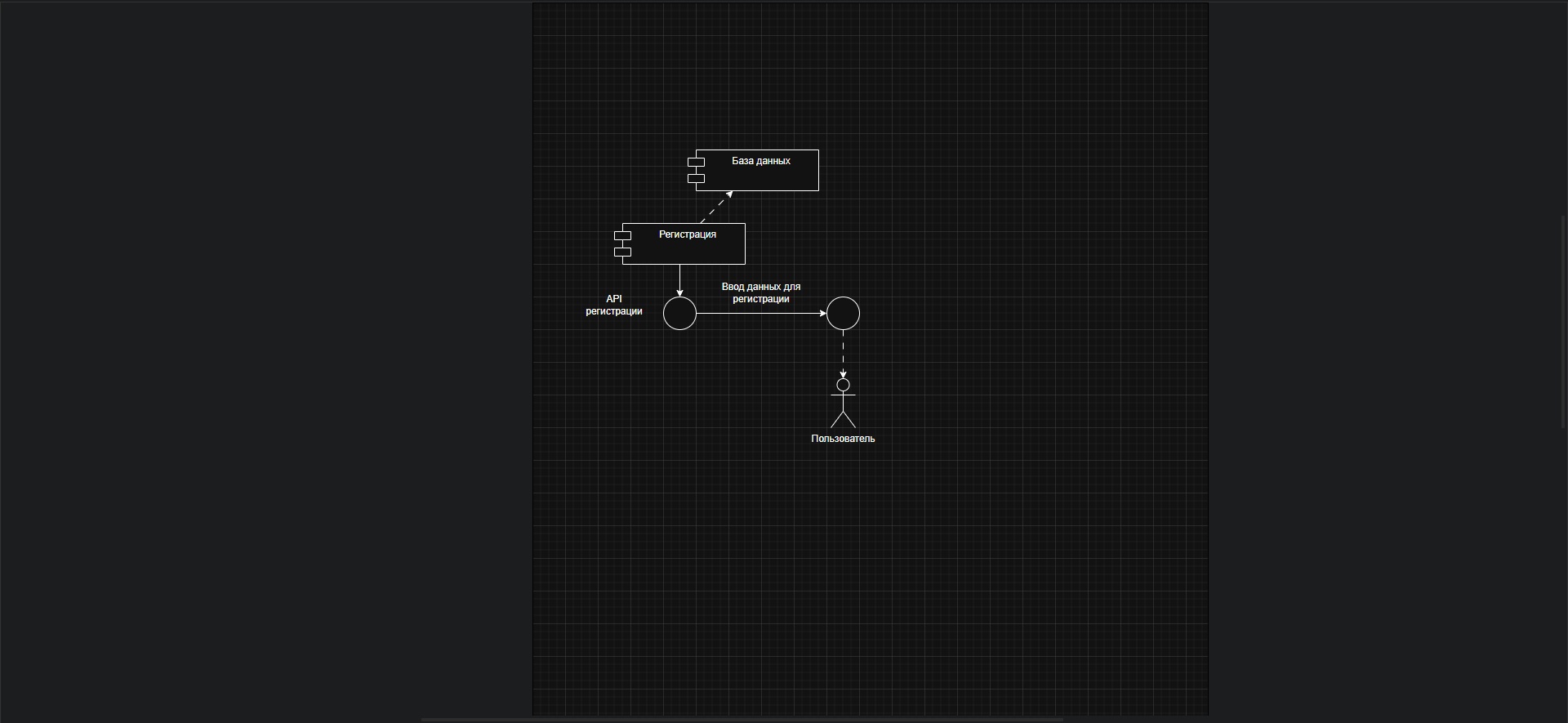
**Рисунок 24. Обновление данных - диаграмма состояния**

****

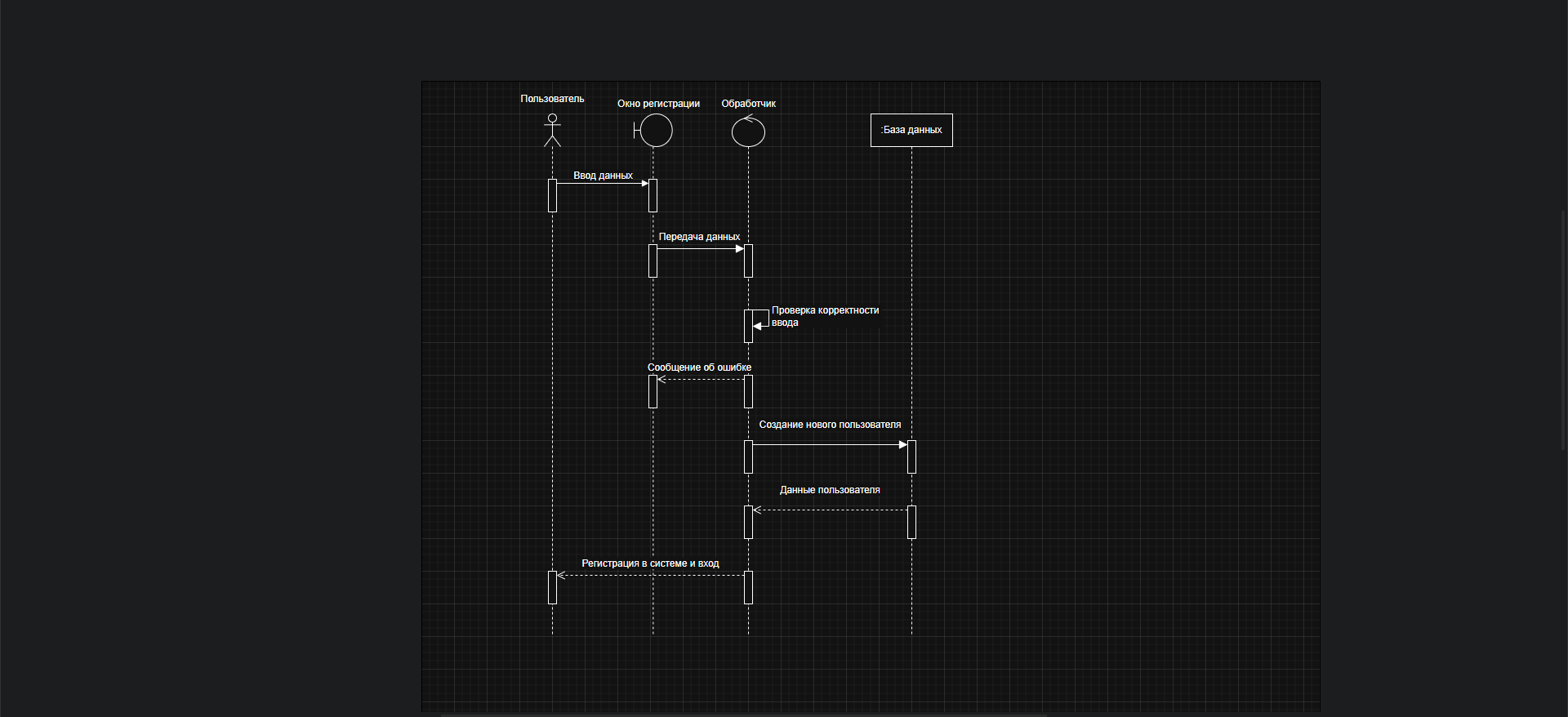
**Рисунок 25. Регистрация - диаграмма кооперации**

****

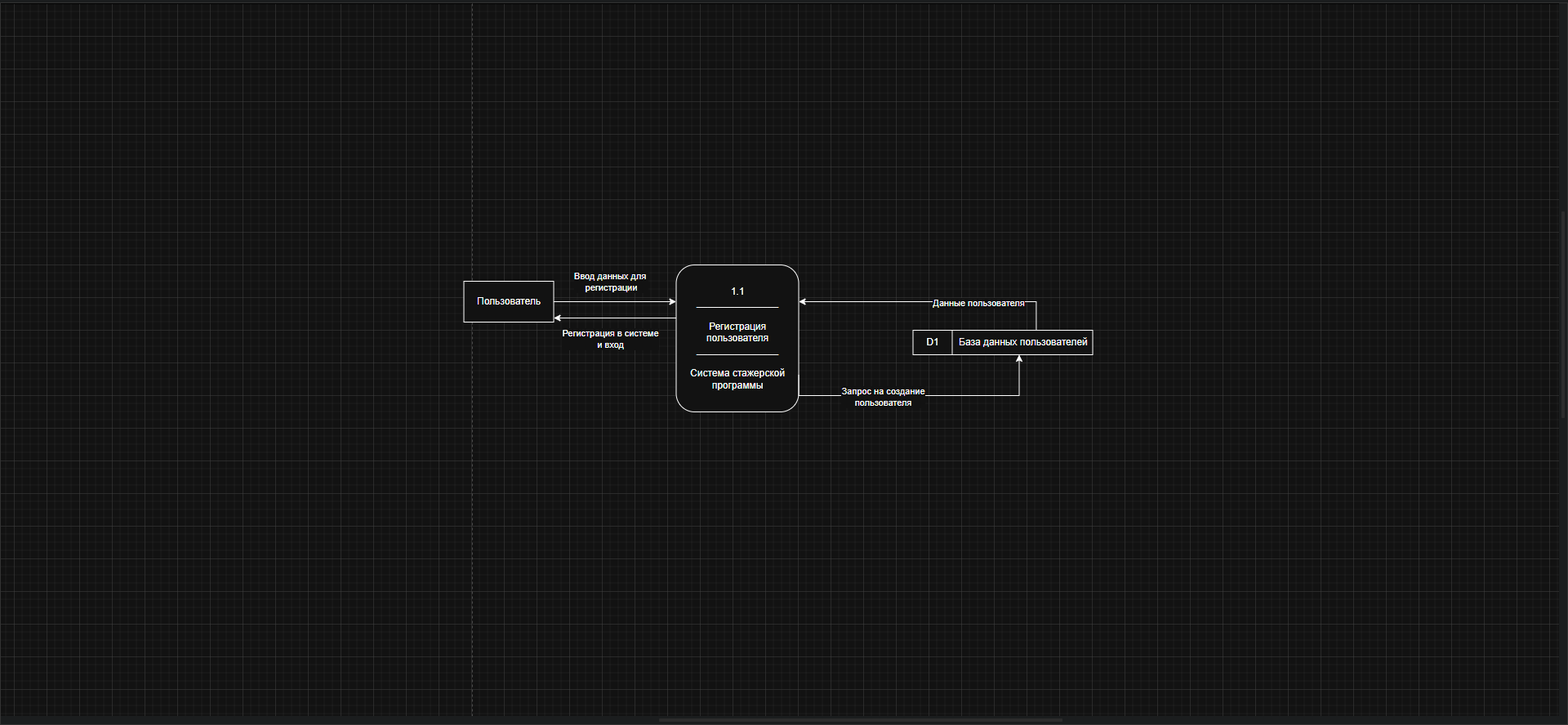
**Рисунок 26. Регистрация - диаграмма деятельности**

****

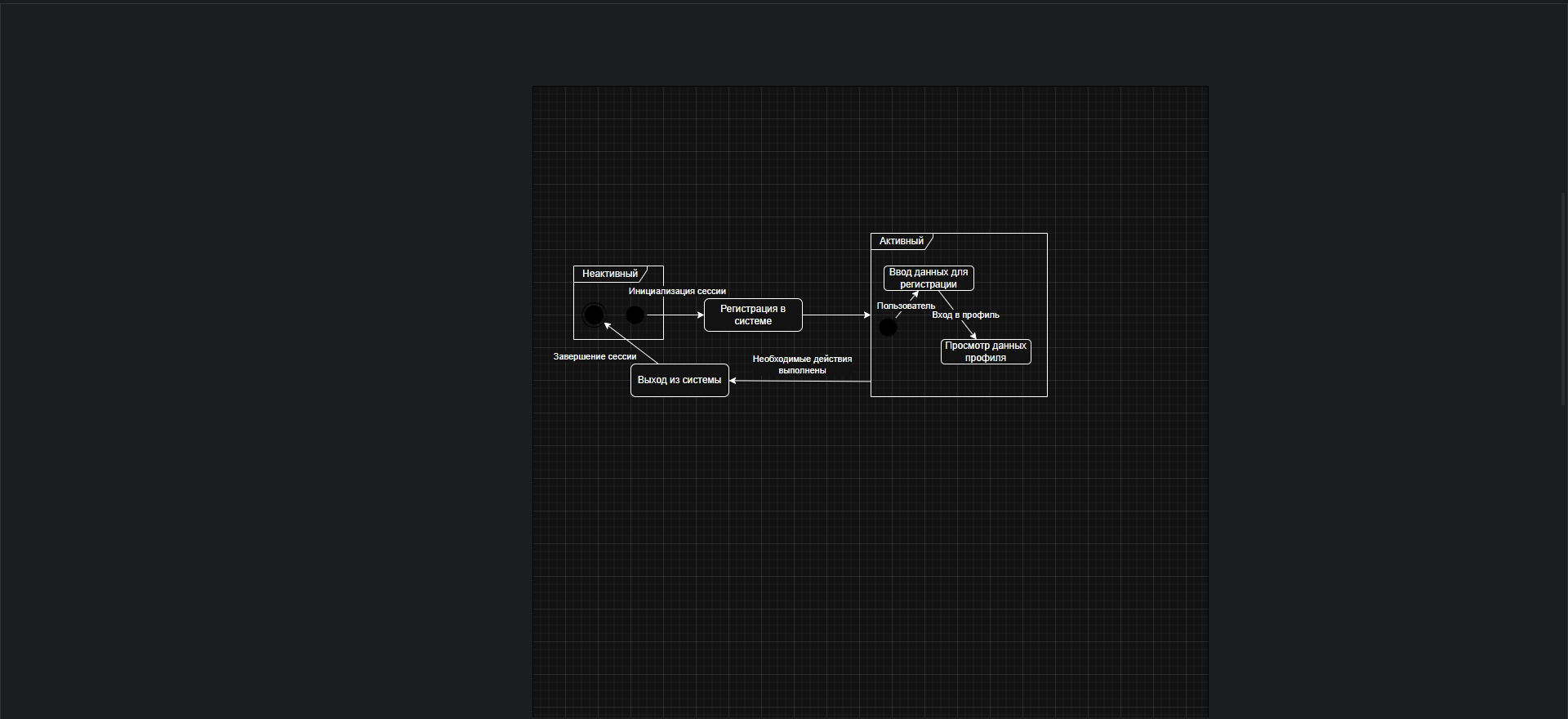
**Рисунок 27. Регистрация - диаграмма компонентов**

****

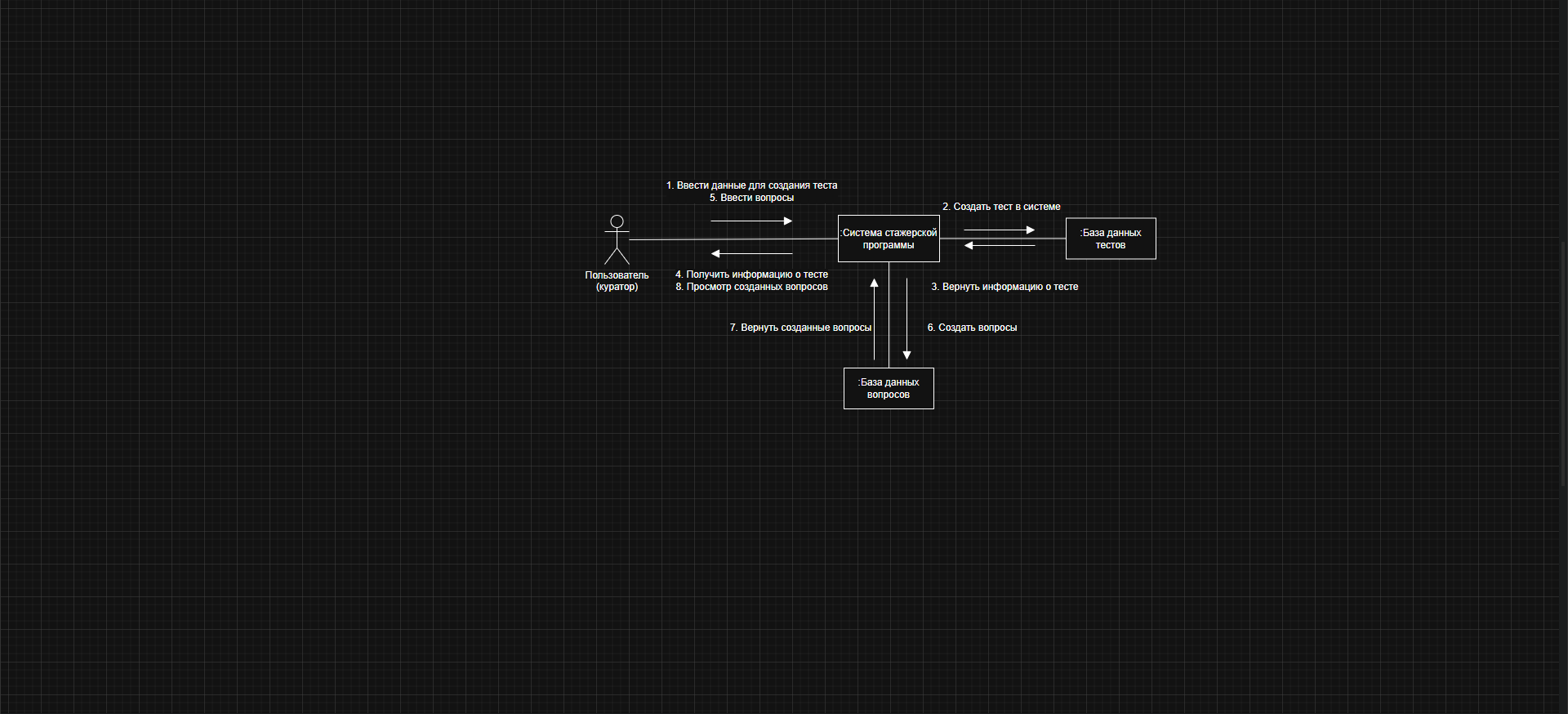
**Рисунок 28. Регистрация - диаграмма последовательности**

****

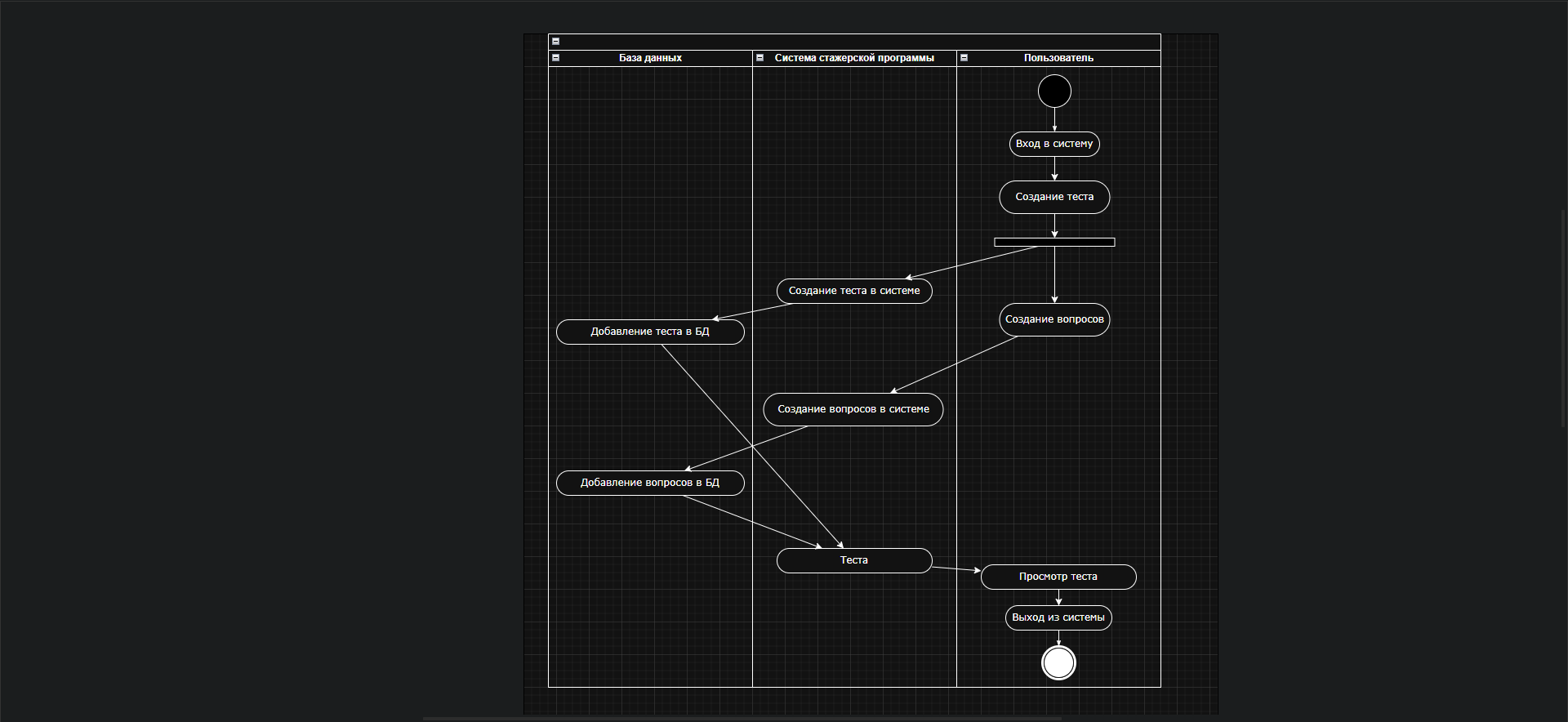
**Рисунок 29. Регистрация - диаграмма потоков данных**

****

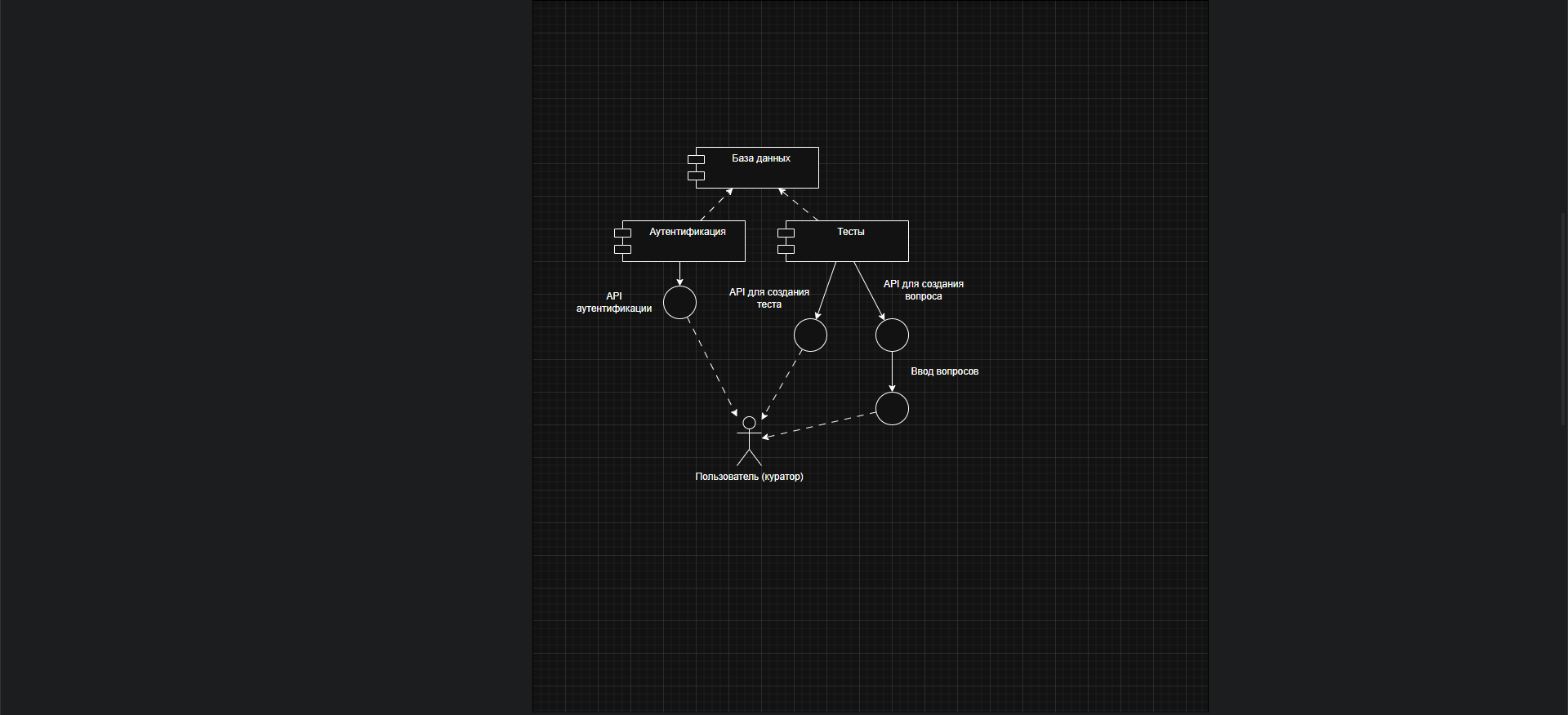
**Рисунок 30. Регистрация - диаграмма состояния**

****

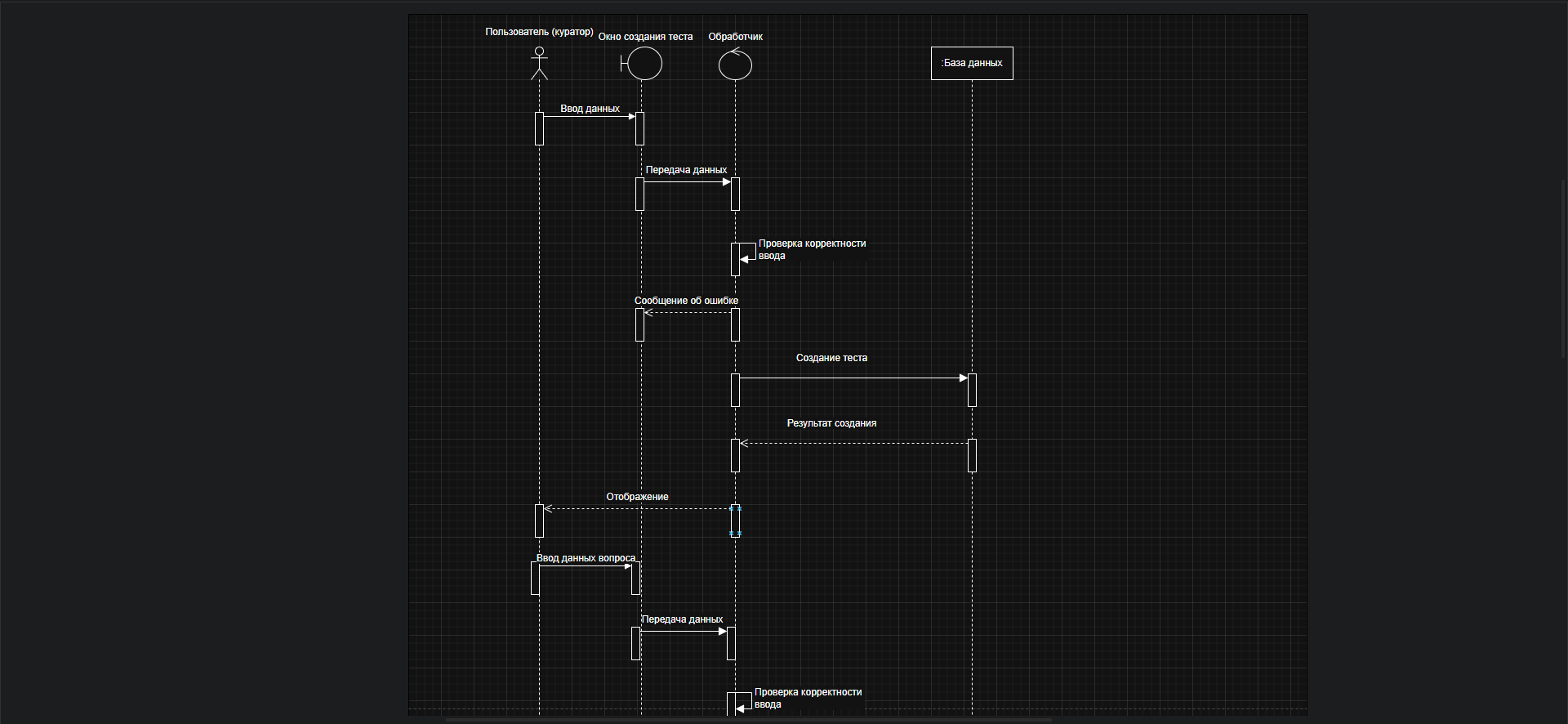
**Рисунок 31. Тесты - диаграмма кооперации**

****

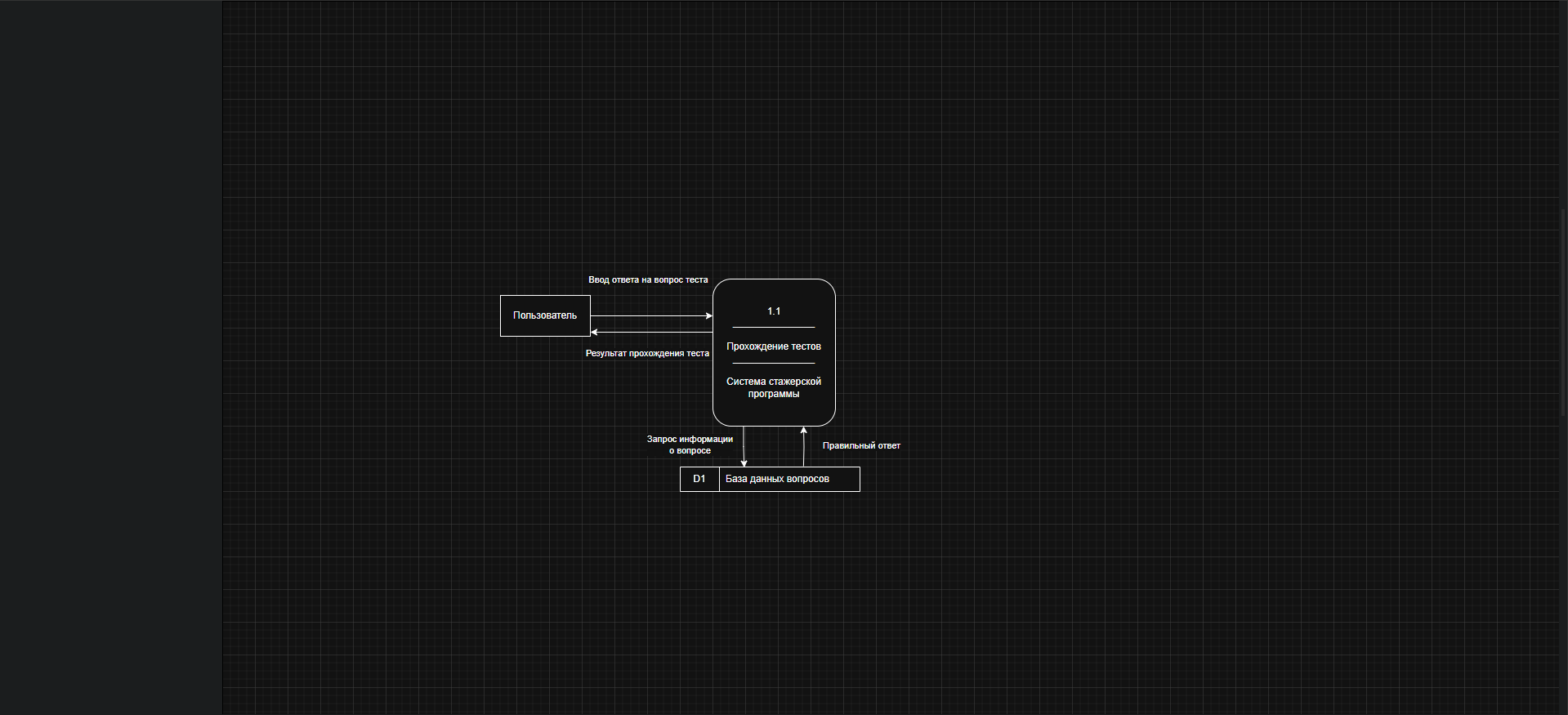
**Рисунок 32. Тесты - диаграмма деятельности**

****

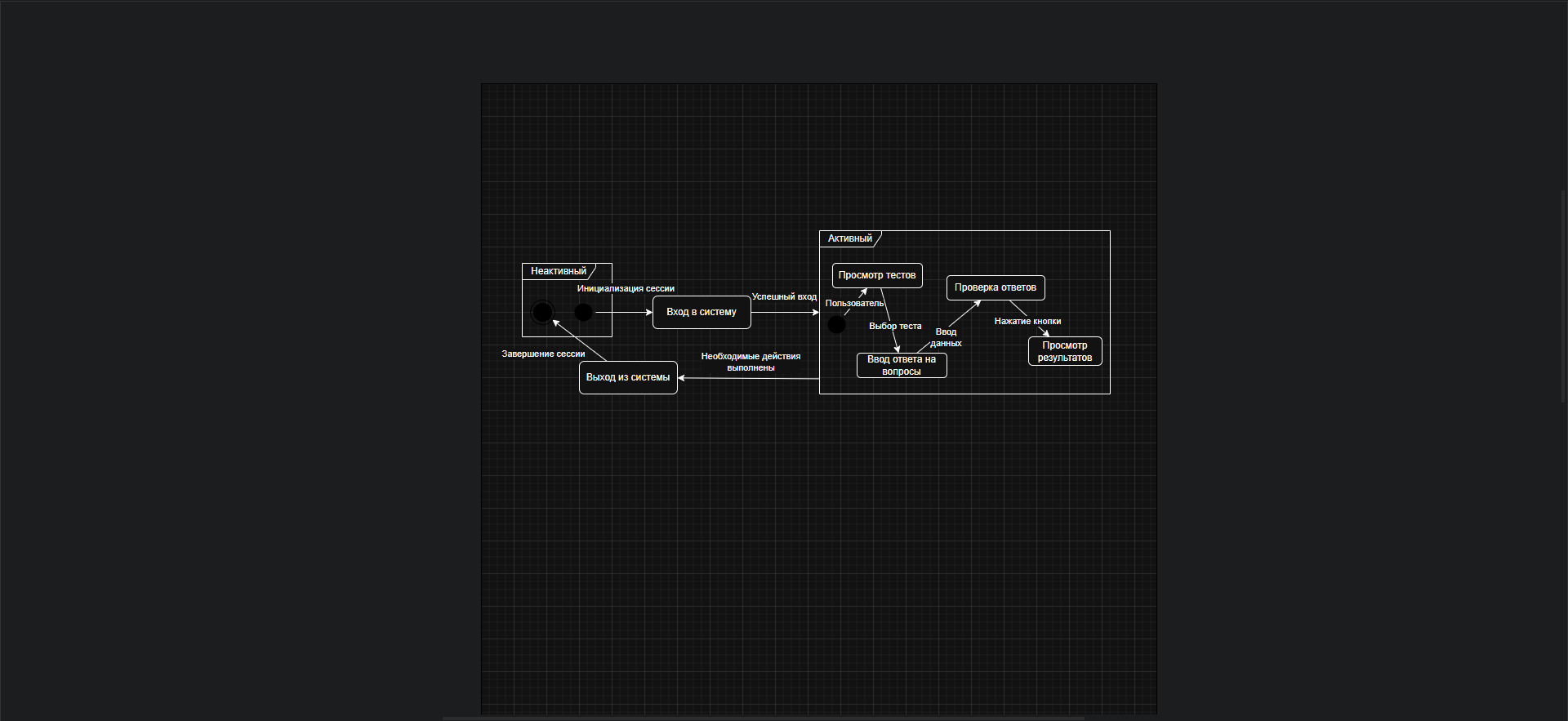
**Рисунок 33. Тесты - диаграмма компонентов**

****

**Рисунок 34. Тесты - диаграмма последовательности**

****

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

****

**Рисунок 36. Тесты - диаграмма состояния**

## **ПРИЛОЖЕНИЕ В**