**ПРИЛОЖЕНИЕ А**

**АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ**

В наши дни очень перспективным и стремительно развивающимся видом предоставления услуг являются грузовые перевозки, а также услуги грузчиков. Фирмы, занимающиеся грузоперевозками, как правило, обладают своим автопарком, размер которого зависит от степени развития фирмы. Фирма, или организация, обладающая своим автопарком и занимающаяся грузоперевозками, называется автотранспортным предприятием.  
Автотранспортные предприятия имеют диспетчерский аппарат, который в грузовых хозяйствах изучает потоки грузов, транспортной связи промышленных предприятий, заключает договоры с грузоотправителями, соблюдает договорные условия; выдаёт сменно-суточные задания шофёрам с учётом максимально возможного использования грузоподъёмности автомобилей и минимального их пробега без груза; организует контроль за работой автомобилей на линии. В отдельных случаях они осуществляют погрузочные и разгрузочные работы, экспедиционные и складские операции, связанные с перевозкой грузов.  
  
В результате анализа предметной области было установлено, что типовой процесс обмена информацией при управлении грузоперевозками  можно представить следующими этапами:  
1. Формирование информации по маршрутам.  
2. Принятие на работу водителя (водителей).  
3. Отражение информации о проделанной работе (то есть о выполненных

перевозках определёнными водителями по определённы  перевозку груза, формирование счёта, выдача заданий водителям путём формирования  
«путевого листа», контроль срока доставки.  
 Задача автоматизации сформулирована в следующем виде:  
В компании, занимающейся перевозками грузов, необходимо отслеживать стоимость перевозок с учетом заработной платы водителей. Компания осуществляет перевозки по различным маршрутам. Для каждого маршрута определено некоторое название, вычислено примерное расстояние и установлена некоторая оплата для водителя. Информация о водителях включает фамилию, имя, отчество и стаж. Для проведения расчетов должна хранится полная информация о перевозках (маршрут, водитель, даты отправки и прибытия). По факту некоторых перевозок водителям выплачивается премия. Классы объектов Маршруты (Название, Дальность, Количество дней в пути, Оплата). Водители (Фамилия, Имя, Отчество, Стаж). Проделанная работа (Маршрут, Водитель, Дата отправки, Дата возвращения, Премия)

Бизнес-процесс включает в себя взаимосвязанные действия, которые реализуют одну (или несколько) из бизнес-целей компании в информационной системе компании. На контекстной диаграмме деятельности транспортной компании изображено функционирование основной бизнес-функции:

− заказы (Вход);

− груз (Вход);

− прибыль (Выход);

− законы РФ (Управление);

− правила и ГОСТы (Управление);

− бухгалтерская система (Механизм);

− сотрудники (Механизм);

− материально-техническое обеспечение (Механизм).

Диаграмма бизнес-процессов в нотации IDEF0 представлена на рисунке 1.

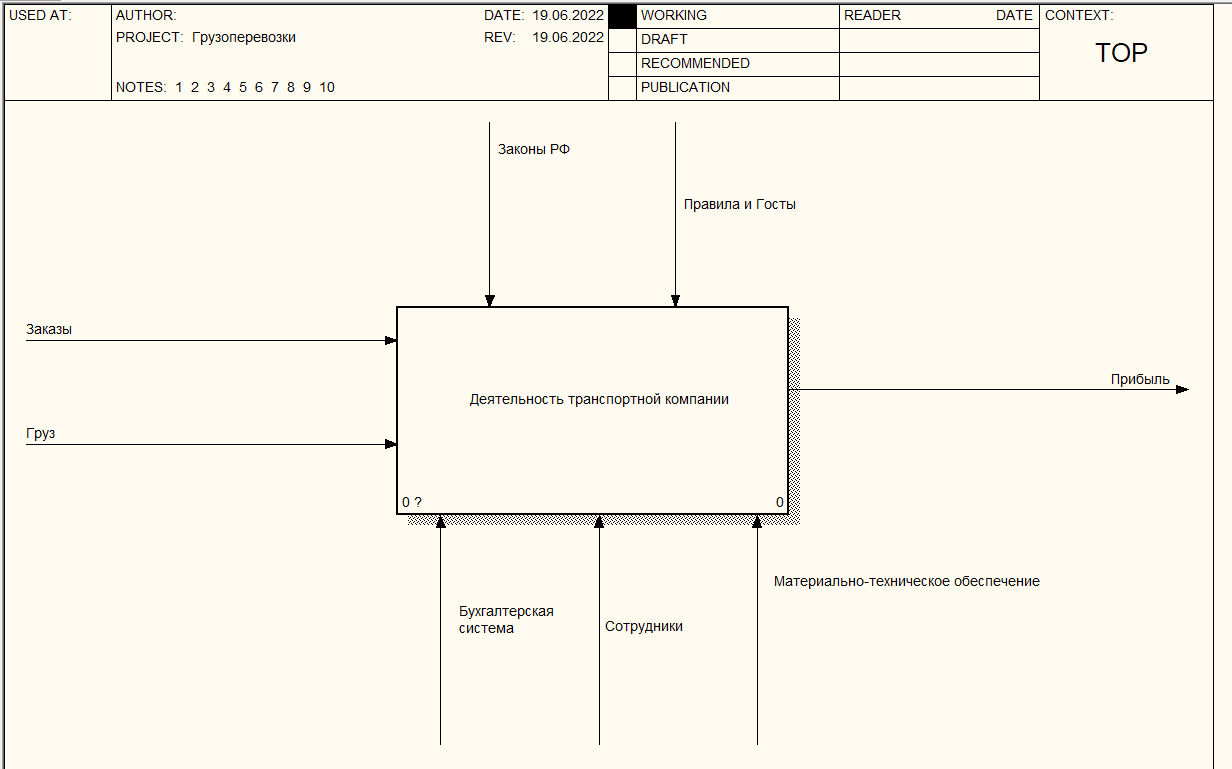


Рисунок 1 – Контекстная диаграмма деятельности

При декомпозиции главной бизнес-функции выделяются следующие функциональные блоки: получение заказа, поиск исполнителя, выполнение заказа. Диаграмма декомпозиции деятельности транспортной компании показана на рисунке 2.

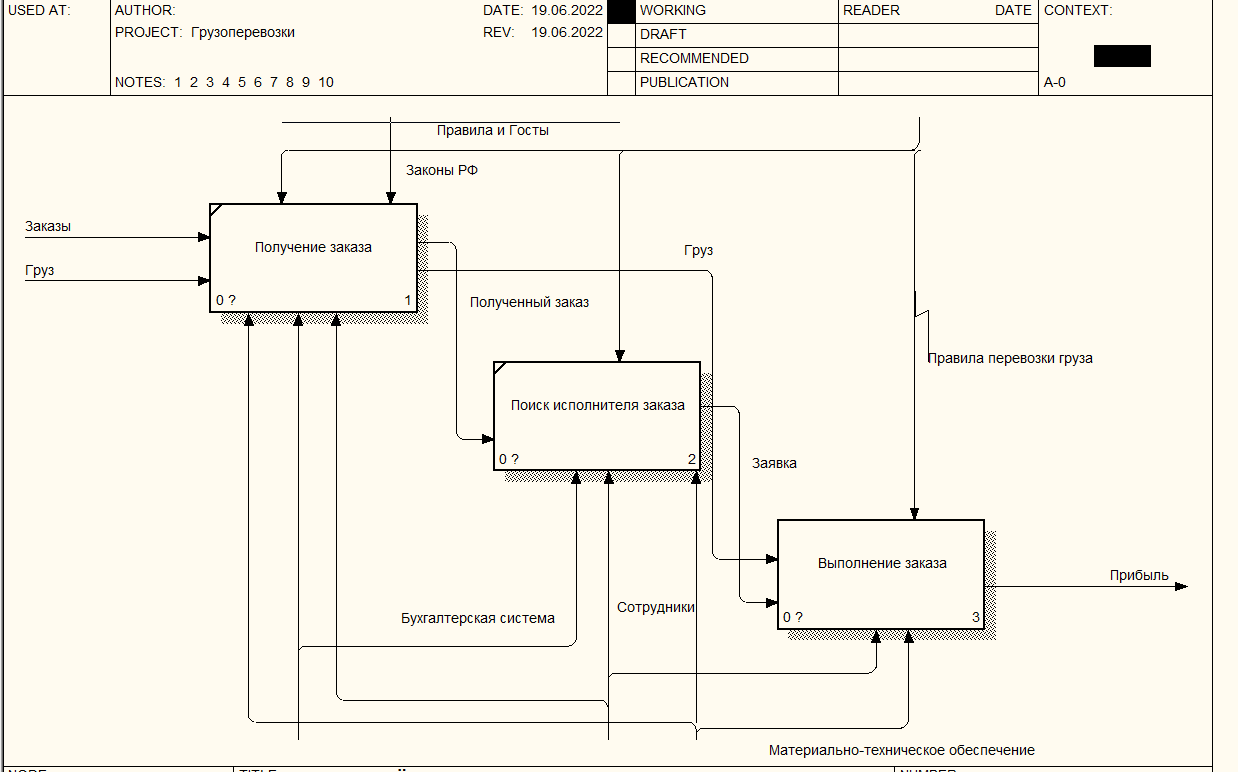


Рисунок 2 – Диаграмма декомпозиции деятельности компании

Декомпозиция выполнения заказа. Сотрудник получает заказ и груз, далее этот груз загружают в транспортное средство. После чего происходит перевозка груза, с учетом правил перевозки грузов. По окончании перевозки, груз выгружается из транспортного средства и происходит оплата выполненного заказа. Диаграмма декомпозиции выполнения заказа показана на рисунке 3.

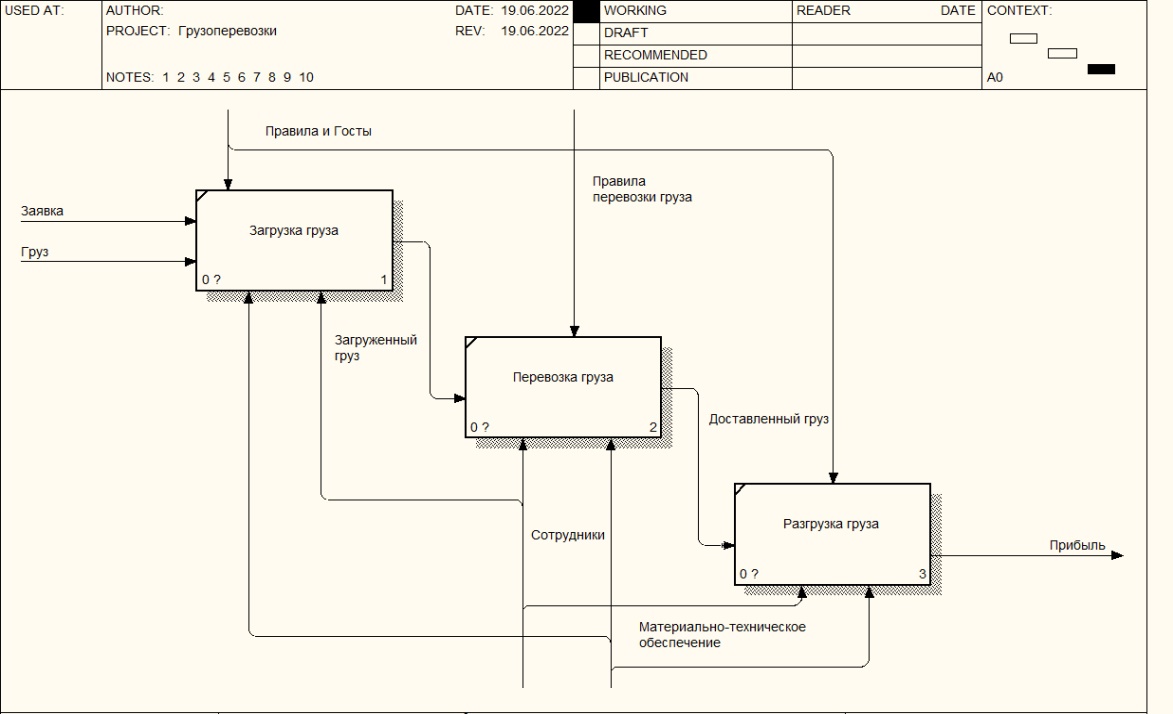


Рисунок 3 – Диаграмма декомпозиции выполнения заказа

Анализ бизнес процессов облегчает понимание работы транспортной организации, и помогает в проектировании и разработке информационной системы.

**ПРИЛОЖЕНИЕ Б**

**ПОСТРОЕНИЕ АРХИТЕКТУРЫ ПРОГРАММНОГО СРЕДСТВА**

1C: Программа проста и удобна в использовании, есть множество языков, большее кол-во функций, гибкая настройка интерфейса, множество различных библиотек, большое кол-во современных инструментов для написания кода, проектирования графический интерфейсов, сборки отладки и тестирования приложений, встроенный поиск и исправления ошибок.

Диаграмма последовательности (англ. sequence diagram) — диаграмма, на которой для некоторого набора объектов на единой временной оси показан жизненный цикл какого-либо определённого объекта (создание-деятельность-уничтожение некой сущности) и взаимодействие акторов (действующих лиц) ИС в рамках какого-либо определённого прецедента (отправка запросов и получение ответов). Используется в языке UML.

Основными элементами диаграммы последовательности являются обозначения объектов (прямоугольники с названиями объектов), вертикальные «линии жизни» (англ. lifeline), отображающие течение времени, прямоугольники, отражающие деятельность объекта или исполнение им определенной функции (прямоугольники на пунктирной «линии жизни»), и стрелки, показывающие обмен сигналами или сообщениями между объектами.

На рисунке 4 изображена диаграмма последовательности для варианта использования «Регистрация заказа**»:**

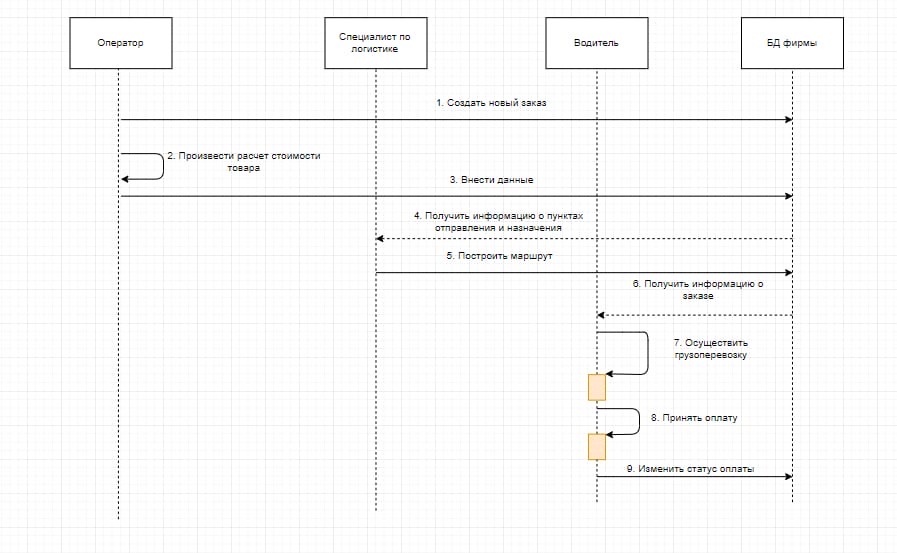


Рисунок 4 -  Диаграмма последовательности для варианта использования «Регистрация заказа»

На данной диаграмме взаимодействия изображена следующая

последовательность событий:

1. заказчик предоставляет необходимые сведения оператору;
2. осуществляет вход в систему;
3. проверяет доступность водителей для грузоперевозки;
4. проверяет доступность автотранспорта.
5. оформляет и регистрирует заказ;
6. подтверждает заказчиком данных заказ;
7. распечатывает накладную для водителя

Моделирование системы будем осуществлять как по уровневый спуск от концептуальной модели к логической, а затем к физической модели системы. Концептуальная модель выражается в виде диаграмм вариантов использования (use-case diagram). Этот тип диаграмм служит для проведения итерационного цикла общей постановки задачи вместе с заказчиком. Диаграммы вариантов использования служат основой для достижения взаимопонимания между программистами-профессионалами, разрабатывающими проект, и заказчиками проекта.

На рисунке 5 изображена диаграмма вариантов использования для проектируемой информационной системы:

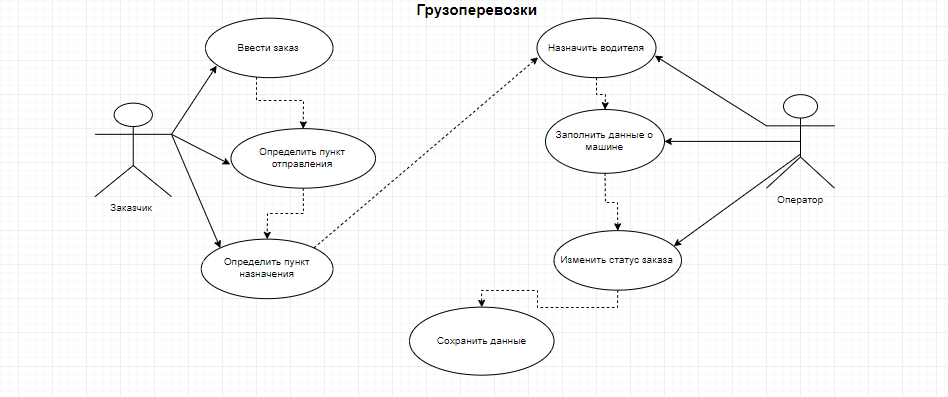


Рисунок 5 - Диаграмма вариантов использования для информационной системы по доставке грузов

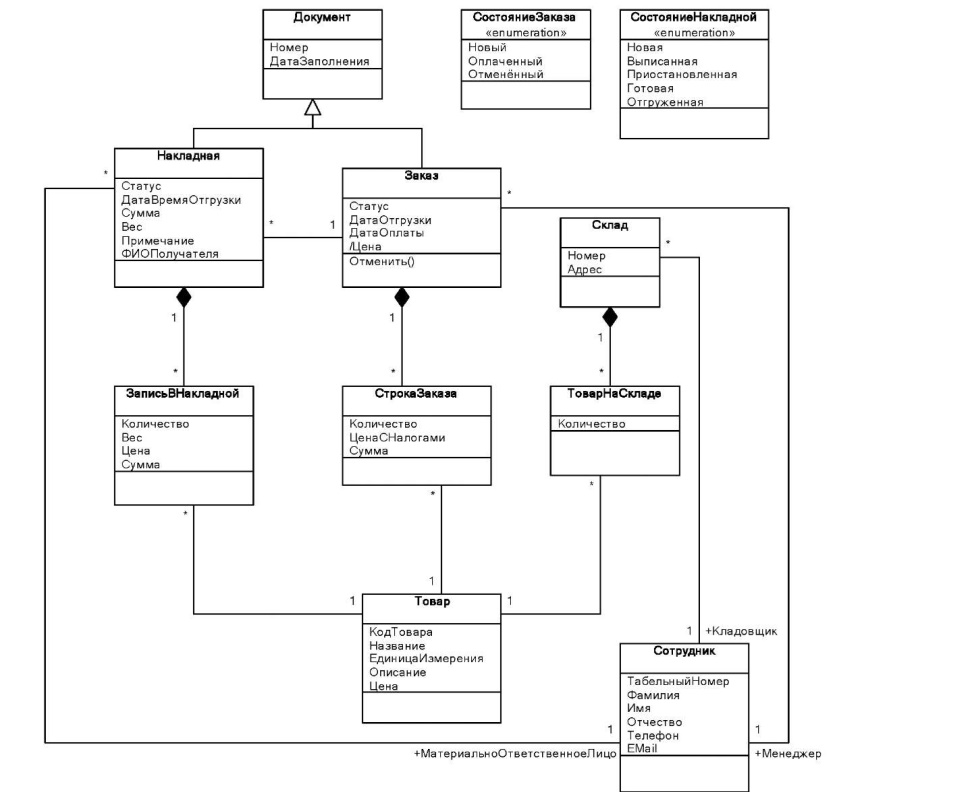


Рисунок 6 – диаграмма классов

**ПРИЛОЖЕНИЕ В**

**РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

Подсистема диспетчеризации реализует логику процессов оформления

заказов на перевозку, распределения по экипажам, формирования маршрутных

листов, контроля выполнения, формирования и обработки путевых листов.

Разнарядка на выпуск автомобилей выписывается с учетом различных

Графиков работы автомобилей и водителей. Программа автоматически

производит проверку пригодности автомобиля к выполнению рейса по

состоянию ТС и документов (истекший полис ОСАГО и т.д.).

При закрытии путевого листа выполняются расчеты нормативного и

фактического расхода ГСМ, выработки ТС, оборудования и водителя, учет отработанного времени для табеля.

В модуле учета ГСМ реализован расчет нормативного расхода топлива в

соответствии с приказом Министерства транспорта России. В справочниках содержатся нормы для многих моделей автомобилей. Для учета отклонений расхода ГСМ от нормы для модели ТС (износ, ремонт и т.д.) можно задать коэффициент для отдельного автомобиля.

Программа формирует различные отчеты и ведомости по движению

ГСМ. Например, есть возможность сформировать ведомость по приходу или расходу, движению ГСМ. Также можно сформовать отчеты по заправкам. На базе введенных данных программа поможет проанализировать поступления

топлива, нормативный и фактический расходы топлива. Есть возможность

сравнения результатов расхода ГСМ на одном ТС под разными водителями.

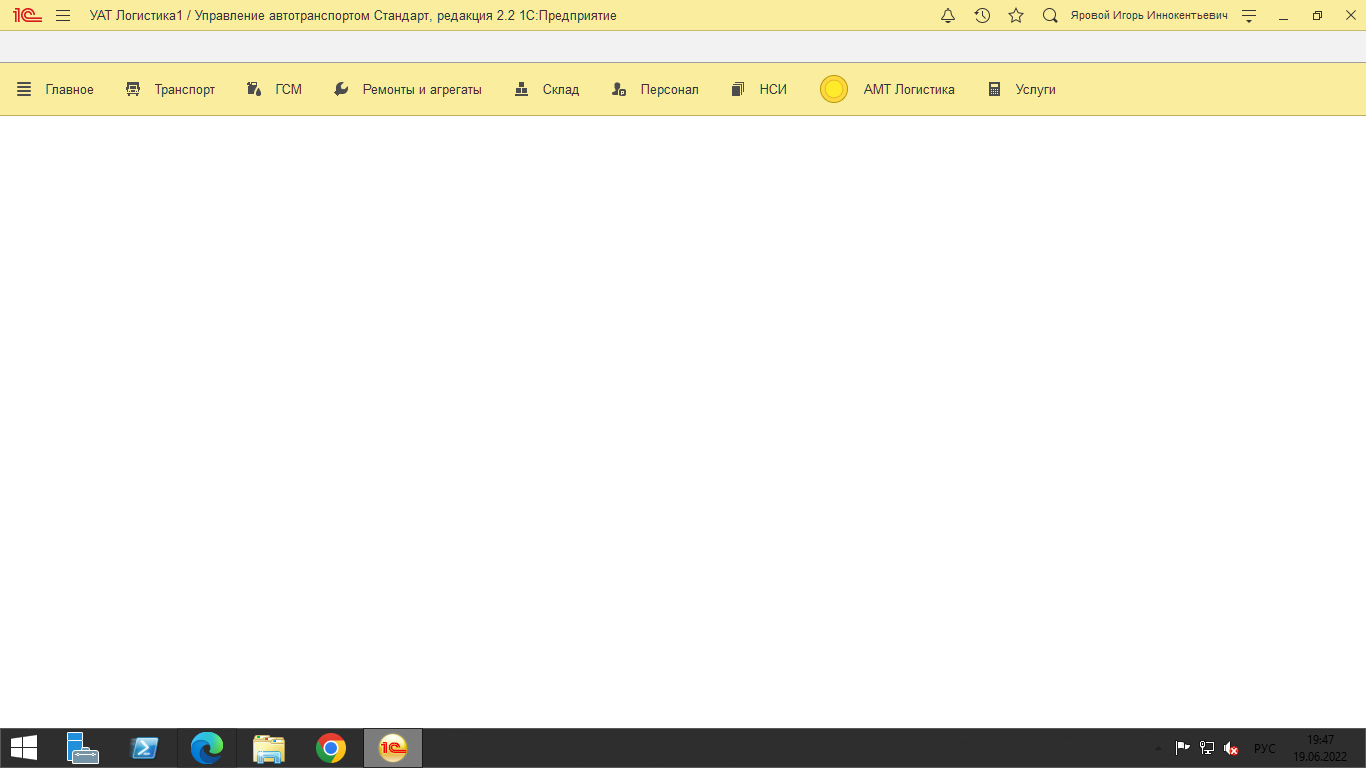


Рисунок 7 – главная страница УАТ Логистика

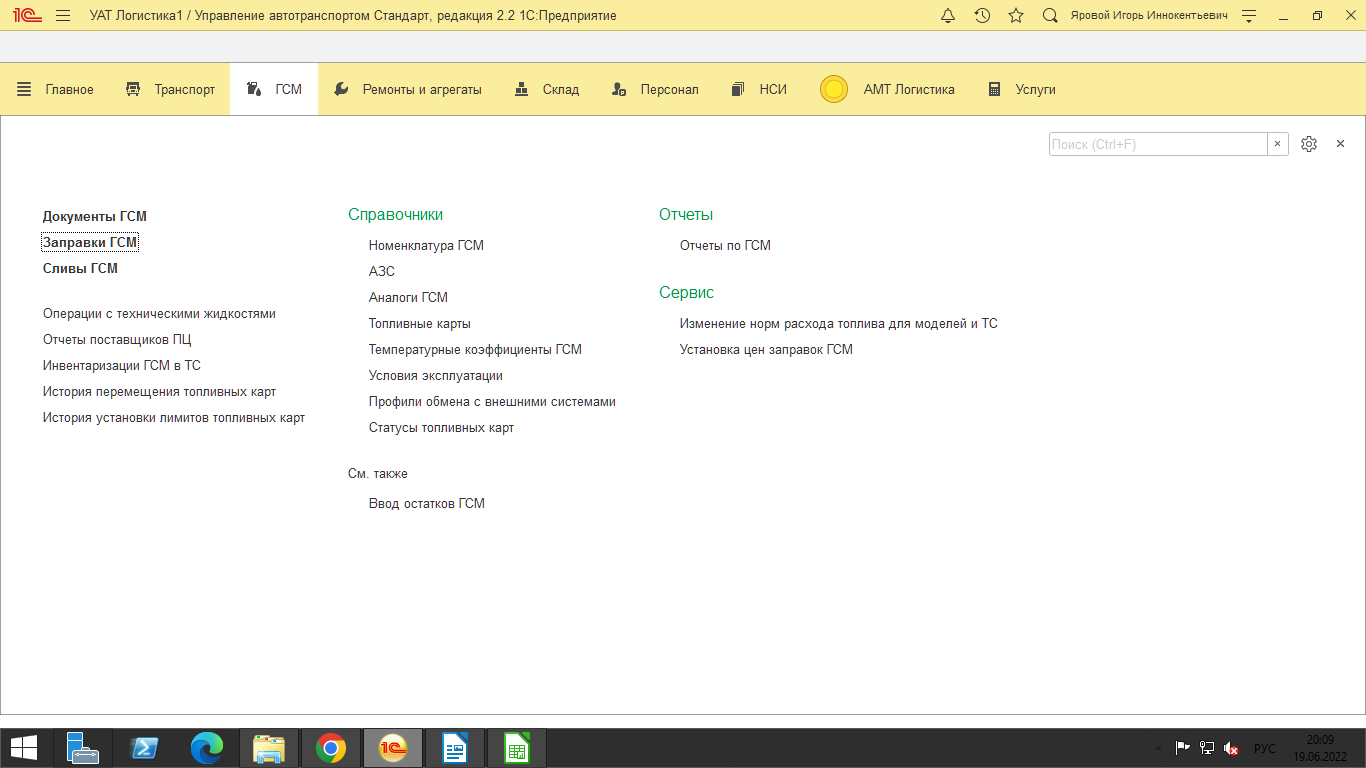


Рисунок 8 – Переход на вкладку ГСМ

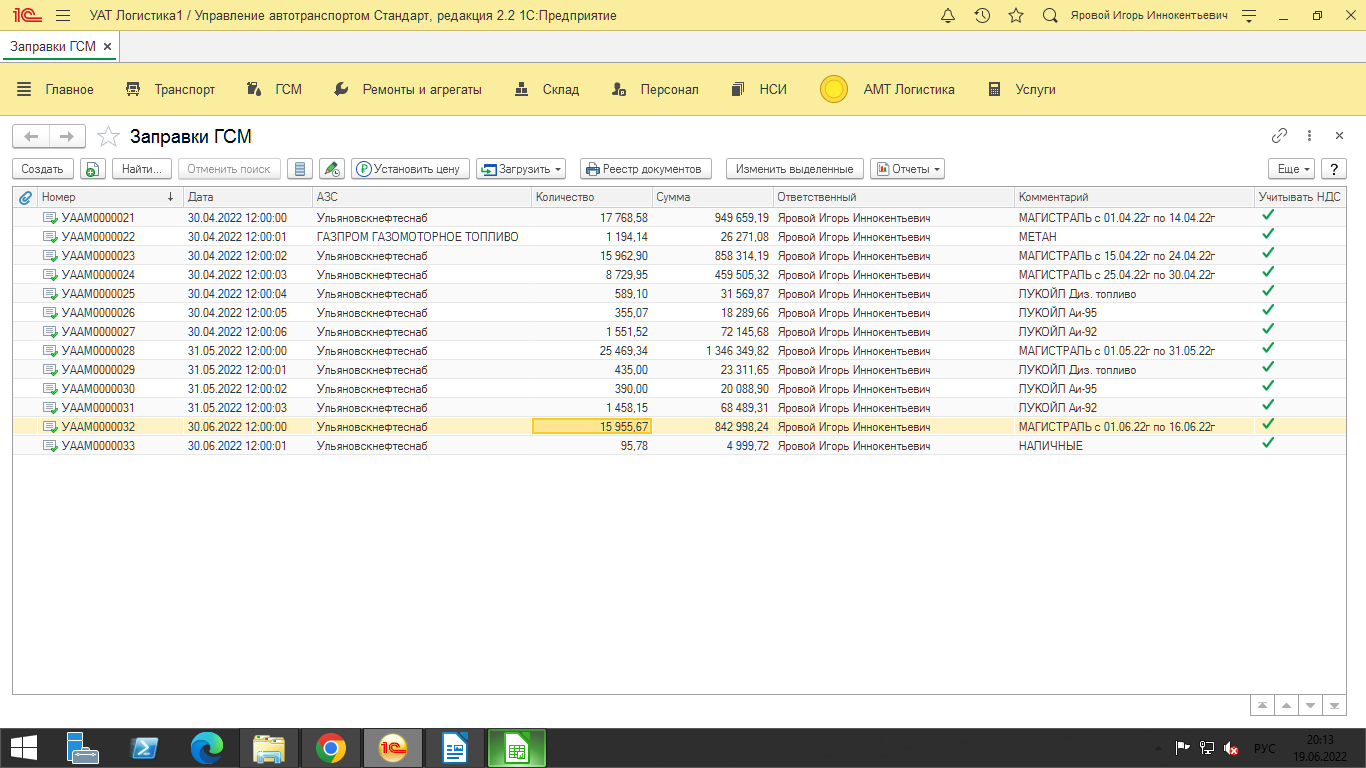


Рисунок 9 – Страница «Заправка ГСМ»

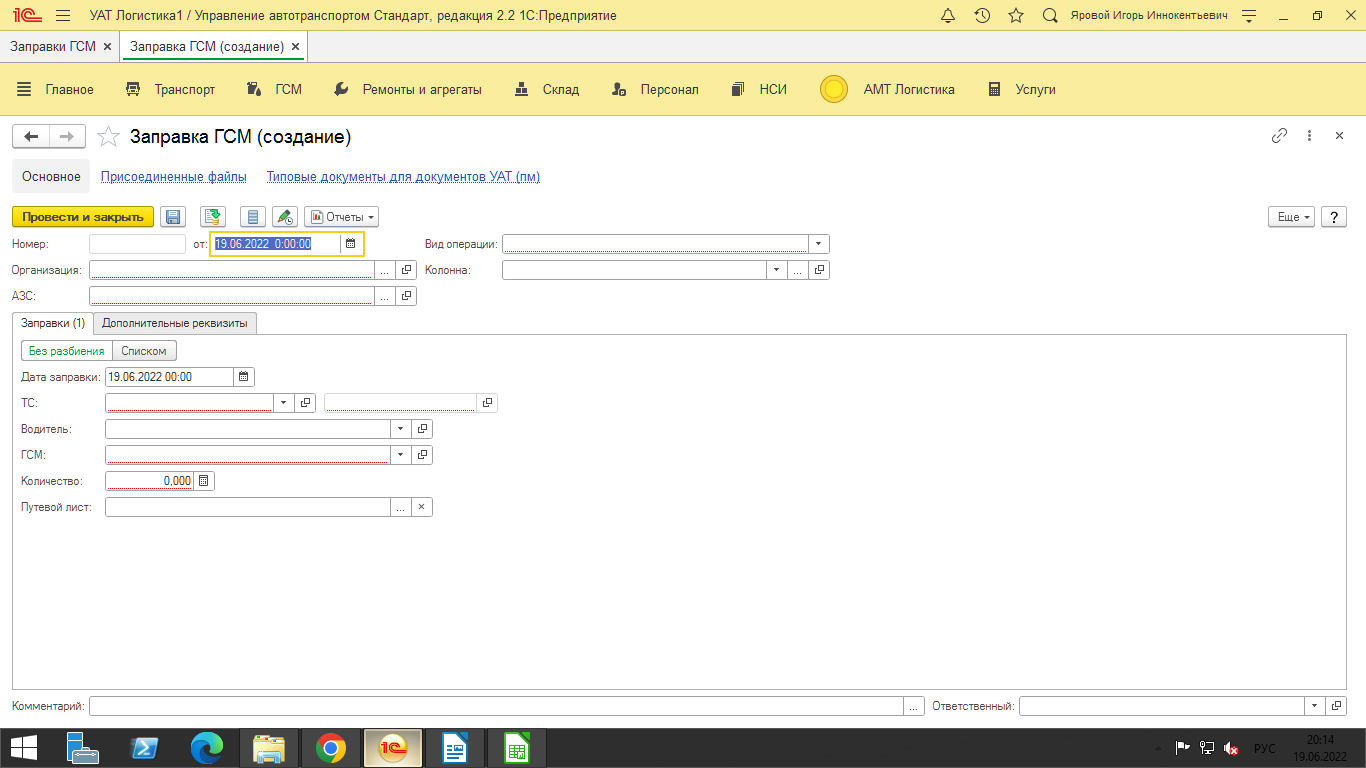


Рисунок 10 – Страница «Заправка ГСМ(создание)»

**ПРИЛОЖЕНИЕ Г.**

**НАСТРОЙКА РАБОТЫ СИСТЕМЫ КОНТРОЛЯ ВЕРСИЙ**

**Описание загрузки проекта в репозиторий**

1. Создали репозиторий в GitHub.
2. Открыли консоль и перещли в директорию проекта
3. Определили текущую директорию, как директорию репозитория.
4. Добавили файлы в ваш новый локальный репозиторий.
5. Коммит файлов мы добавили в коммит в наш локальный репозиторий.
6. Скопировали ссылку на наш репозиторий, и добавили в командной строке.

**Централизованные системы контроля версий**



# Рисунок 11– система контроля версий

Такой подход имеет множество преимуществ, особенно перед локальными СКВ. Администраторы имеют полный контроль над тем, кто и что может делать, и гораздо проще администрировать ЦСКВ, чем оперировать локальными базами данных на каждом клиенте. Система защищает исходный код от потери. В данном проекте используется единственный сервер, содержащий все версии файлов, и некоторое количество клиентов, которые получают файлы из этого централизованного хранилища.

**ПРИЛОЖЕНИЕ Д.**

**ТЕСТИРОВАНИЕ ИНФОРМАЦИОННОЙ СИСТЕМЫ**

**Автоматизированное тестирование**

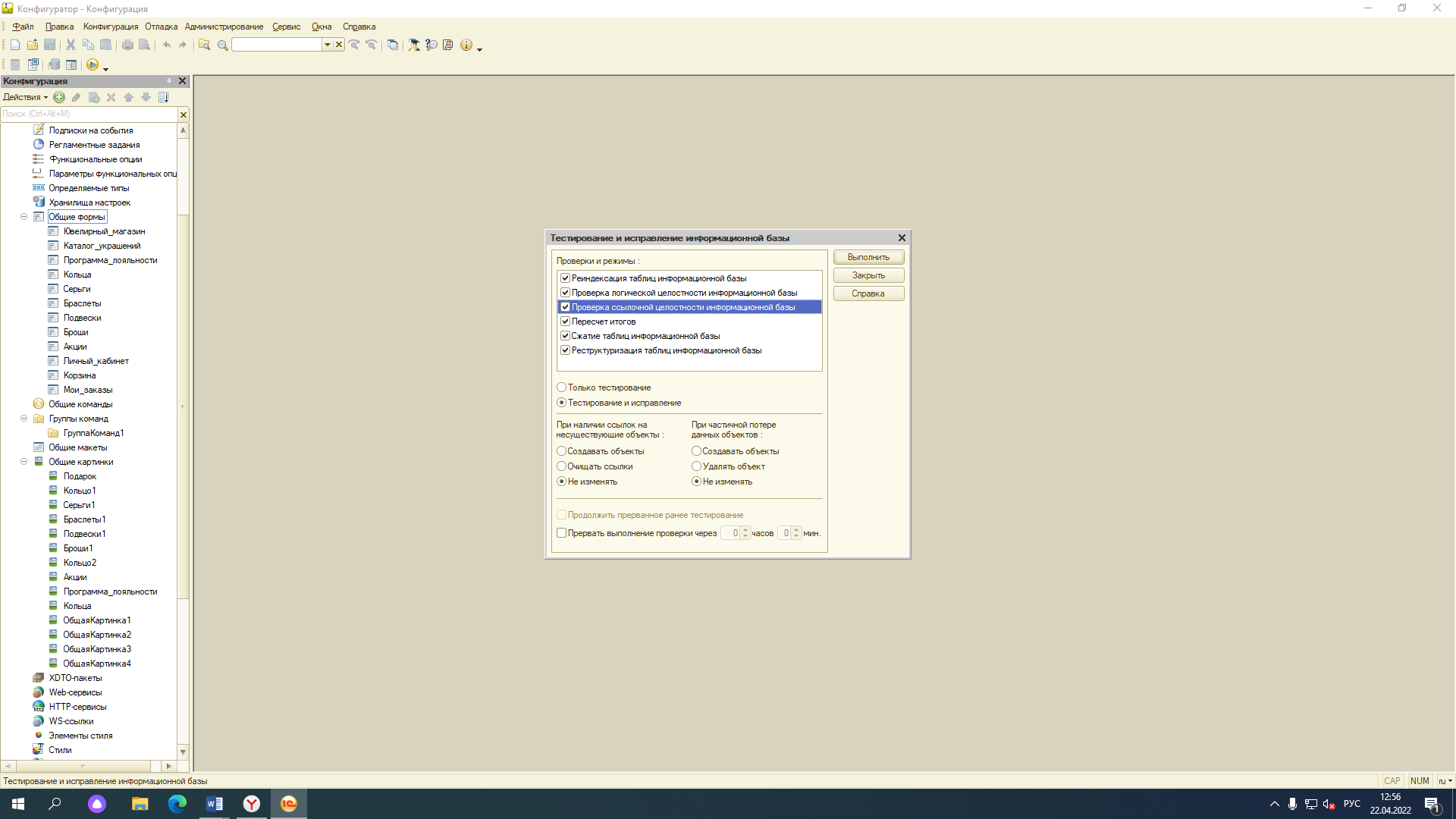


Рисунок12 - Автоматизированное тестирование

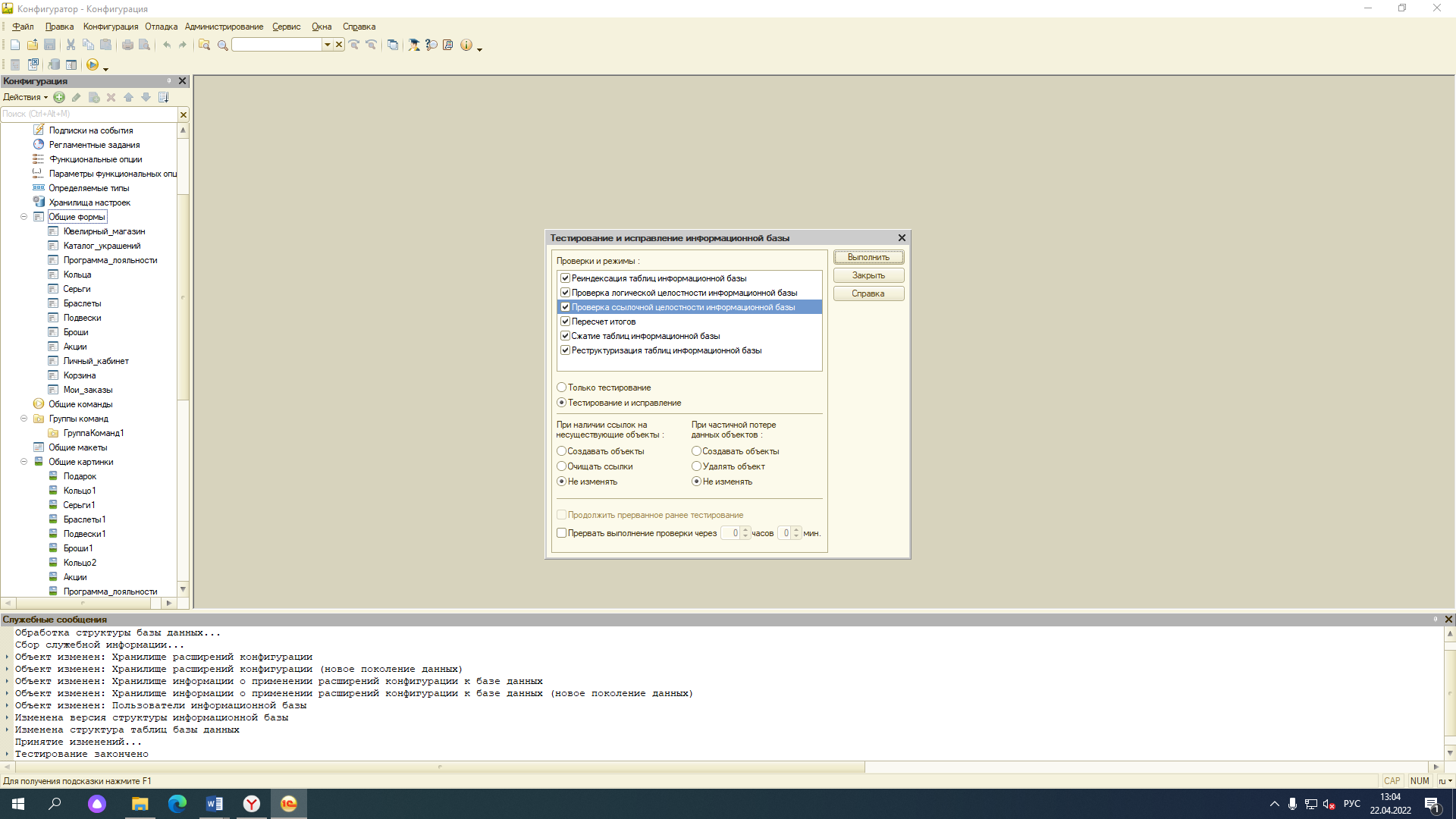


Рисунок 13 - Результат автоматизированного тестирования