

Содержание

Введение.....	3
1 Постановка задачи.....	6
1.1 Описание предметной области.....	6
1.2 Функции проектируемой информационной системы.....	8
2 Разработка UML-диаграмм этапа проектирования.....	10
3 Разработка инфологической схемы базы данных.....	13
4 Программные средства для разработки ИС для предметной области «Предприятие автоперевозок».....	14
5 Проектирование интерфейса приложения.....	17
6 Программа и методика испытаний информационной системы.....	20
7 Создание динамической Web-страницы. Понятие объекта, функции, метода и события. Web-службы. Понятие динамической Web-страницы.....	23
8 Способы разработки Web-страниц. Языки гипертекстовой разметки.....	27
Заключение.....	29
Список использованной литературы.....	30
Приложение А.....	31
Приложение Б.....	38

Введение

Автомобильные перевозки - один из недорогих способов транспортировки. Этот тип грузоперевозок подходит для обеспечения товаром городских торговых точек, для доставки из города в районы, а также для перемещения внутри страны.

Конкуренция среди логистических компаний требует постоянного повышения эффективности услуг. При наличии такой разветвленной транспортной системы как в России крайне важно профессионально составлять маршруты перевозок. Чтобы обеспечить высокий уровень сервиса, сделать работу более оперативной и качественной, необходимо автоматизировать все процессы. Значит, необходимо решение в виде специализированной информационной системы.

В настоящее время существует целый ряд программ для транспортных компаний. Однако они не всегда могут обеспечить требуемую функциональность.

В транспортную компанию поступает большое количество заявок от постоянных клиентов и разовых пользователей услугами перевозок. С постоянными клиентами ведутся проекты. Для эффективной работы компании необходима такая информационная система (ИС), которая помогала бы менеджеру работать с большим количеством информации и была удобной для управления процессами грузоперевозок.

Процесс управления грузоперевозками включает в себя: обработку заявки клиента, планирование маршрута, планирование работы водителей, планирование работы автотранспорта, планирование консолидации грузов (объединение грузов), составление отчетов. Все перечисленные процессы должны быть включены в информационную систему. Её можно разбить на модули:

- управление грузоперевозками (планирование рейсов, работы водителей)

и автотранспорта, контроль доставки);

- отчетность (анализ эффективности работы компании со всех сторон, отчеты различных видов).

Автоматизация составления маршрута транспортировки сложна в реализации. Чтобы найти оптимальный путь перевозки груза, требуется учесть множественные факторы, которые имеют непостоянный характер. Например, один из важнейших параметров - загруженность дорог. Особенно это актуально в крупных городах. Другая проблема, связанная с загруженностью, - время составления маршрута.

В корпоративной ИС хранится огромное количество информации. Для принятия решений, анализа работы, контроля труда необходимо правильное представление данных. Базы данных ориентированы в первую очередь на компактное и непротиворечивое хранение информации, а не на понятное представление данных. Вследствие этого информация, хранящаяся в корпоративных ИС, как правило, используется неэффективно. Главная проблема в ИС - не хранение информации, а предоставление ее конечному пользователю в виде отчета в нужном контексте. Составление отчетов - рутинная задача менеджера, занимающая много времени, требующая концентрации внимания и ответственности. Отчеты могут быть предназначены для предоставления клиенту, для бухгалтерии или руководства. Также они могут стать незаменимыми помощниками в работе менеджера-логиста.

Цель работы – рассмотреть теоретические и практические аспекты процесса организации автоперевозок транспортной компании и разработать информационную систему. Разрабатываемая система должна автоматизировать процесс хранения и обработки информации о перевозках. Основной целью является создание и поддержка интерфейса взаимодействия рабочего места логиста, обеспечивающего хранение и обработку информации с помощью клиентского приложения, и базы данных.

Проектируемая система должна использоваться для хранения информации о перевозчиках, заказчиках и загрузках.

Система создается для обслуживания логиста организации – он сможет контролировать процесс выполнения заказа, вести учет водителей и заказчиков.

Практическая значимость состоит в программной реализации информационной системы, позволяющей вести учёт информации по водителям и осуществляемым ими перевозкам. Практические результаты могут быть использованы для применения в компании, осуществляющей автомобильные грузоперевозки с целью увеличения эффективности планирования перевозок, контроля сроков их осуществления.

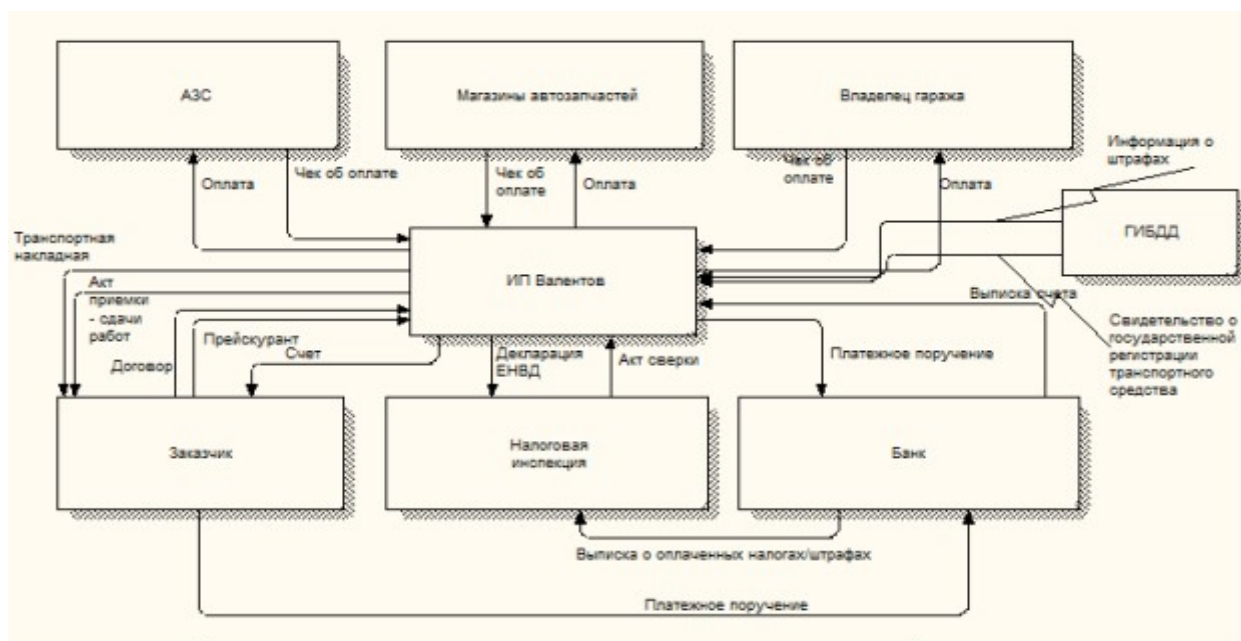
1 Постановка задачи

1.1 Описание предметной области

Объектом исследования является автоматизация процессов учета и анализа деятельности грузоперевозок индивидуальным предпринимателем Валентовым А.В. Это предприятие ориентированно на удовлетворение потребности в транспортировке большого количества грузов, на значительном расстоянии превышающим городские границы. Основа клиентской базы представлена преимущественно крупными организациями, в частности компания «ЮСИЛ».

Структура предприятия представлена в лице Валентова А.В. данный предприниматель взаимодействует с различными организациями представленных заказчиками, а также с различными коммерческими и некоммерческими организациями. Схема документооборота представлена на рисунке 1.

Рисунок 1 – Схема документооборота



Разрабатываемый продукт служит с целью повышения возможностей мониторинга производственных и экономических показателей, по средствам учета и анализа деятельности предприятия. Основными фактором успешности предприятия будут являться: количество выполненных заказов и объем прибыли в определенный промежуток времени. Деятельность предприятия будет считаться успешной в том случае если количество выполненных заказов не будет уменьшаться и разница между доходами и расходами будет больше нуля.

Субъектами документооборота являются:

- АЗС – организация, поставляющая ГСМ;
- магазины автозапчастей – различные дистрибьюторы автозапчастей;
- владелец гаража – лицо предоставляющее в аренду помещение для хранения транспортного средства;
- ИП Валентов – предприниматель учет деятельности которого нуждается в автоматизации;
- ГИБДД – государственная служба, контролирующая соблюдение ПДД;
- заказчик – лицо, осуществляющее заказ на перевозку груза.
- налоговая инспекция – государственная организация по контролю за деятельностью предприятий на территории России, осуществляет сбор налоговых отчислений;
- банк – финансовая организация регулирующая обращение денежных средств между субъектами.

Документами документооборота являются:

- оплата – денежные средства;
- чек об оплате – документ, подтверждающий факт оплаты;
- информация о штрафах – документ, информирующий субъекта о штрафных санкциях;
- свидетельство о государственной регистрации транспортного средства - свидетельство, предоставляемое предпринимателю о постановке транспортного средства на учет;

- транспортная накладная – документ, предназначенный для учета движения товарно-материальных ценностей и расчетов за их перевозки автомобильным транспортом;

- акт приемки – сдачи работ – документ, подтверждающий фактическое оказание комплекса работ, связанных с перевозкой груза, подписывается обеими сторонами;

- договор – документ, регулирующий отношения между заказчиком и исполнителем.

- прейскурант – документ регулирующий ценовую политику отношений между заказчиком и исполнителем.

- счет – Выставляется заказчику за выполненную работу;

- декларация ЕНВД – форма отчетности в налоговую;

- акт сверки – двусторонняя сверка расчетов и оказанных услуг, за определенный период времени;

- платежное поручение – документ, содержащий указание банку о перечислении денежных средств со счета плательщика на счет получателя;

- выписка счета – банковский документ, прилагающийся к пакету документов, отражает поступление/выбытие с расчетного счета в банке;

- книга учета – сводный учет расходов всех хозяйственных операций по различным категориям;

- Выписка о оплаченных налогах/штрафах – банковский документ, подтверждающий оплату налогов/штрафов от хозяйствующего субъекта в налоговую.

1.2 Функции проектируемой информационной системы

Информационная система будет обеспечивать:

- ввод, удаление и редактирование той или иной информации, с возможностью автоматизации (подстановок из других таблиц, фиксированных наборов данных и т.д.) и входного контроля от некорректных действий.

- ограничение доступа к ИС с помощью идентификации пользователя (парольная защита).
- осуществлять поиск по разнообразным признакам (не менее одного в каждой из информационных категорий, например, ФИО водителя и т.д.).
- осуществлять сортировку по различным признакам (не менее одного в каждой из информационных категорий, например, по характеристикам перевозимых грузов и т.д.).
- выводить информацию по всем водителям, которые совершали перевозки грузов за выбранный промежуток времени в заданном направлении.
- вывод информации о том, кто за заданный промежуток времени совершил максимальные (в километровом исчислении) перевозки.
- выводить информацию для любого заданного водителя, о всех маршрутах, в которые он ездил в текущем месяце.

2 Разработка UML-диаграмм этапа проектирования

На рисунке 2 представлена диаграмма активности разрабатываемой системы, представляющая переходы потока управления от одной деятельности к другой. Диаграмма деятельности моделирует жизненный цикл объекта, с переходами из одного состояния (деятельности) в другое.

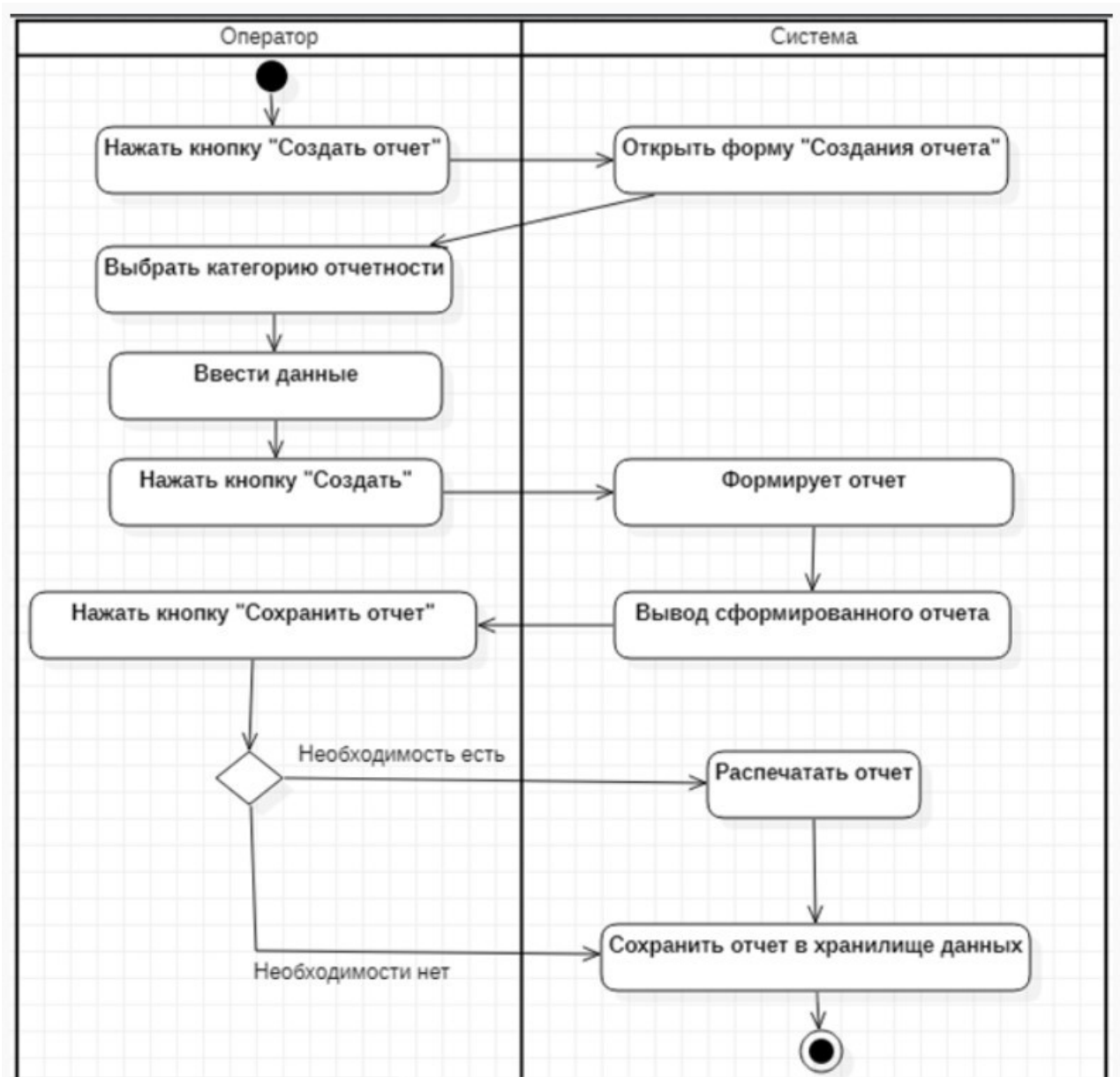


Рисунок 2 - Диаграмма активности

На рисунке 3 представлена диаграмма пригодности, отображающая

объекты, участвующие в сценарии, и их взаимодействие.

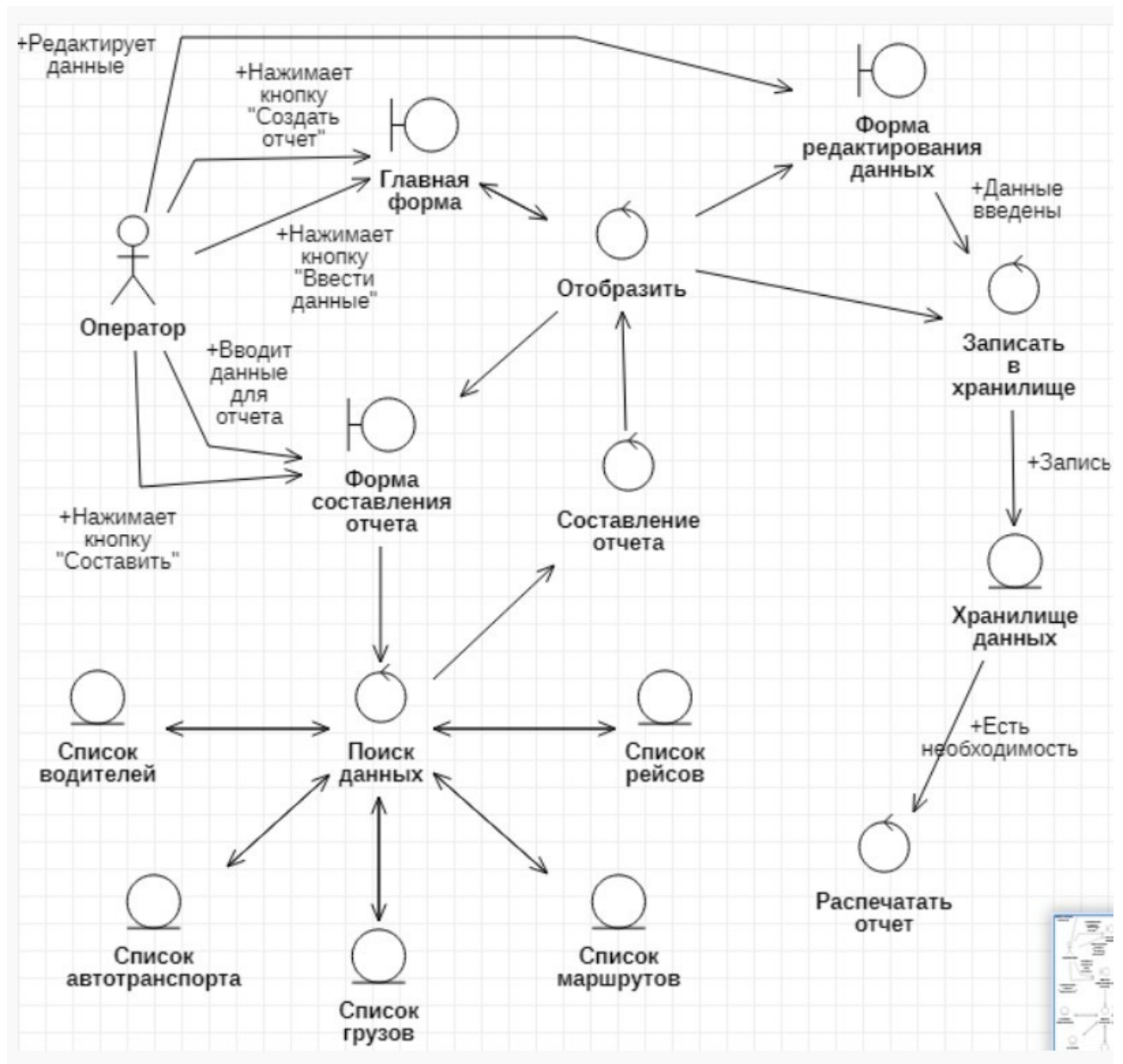


Рисунок 3 - Диаграмма пригодности

На рисунке 4 представлена диаграмма последовательности.

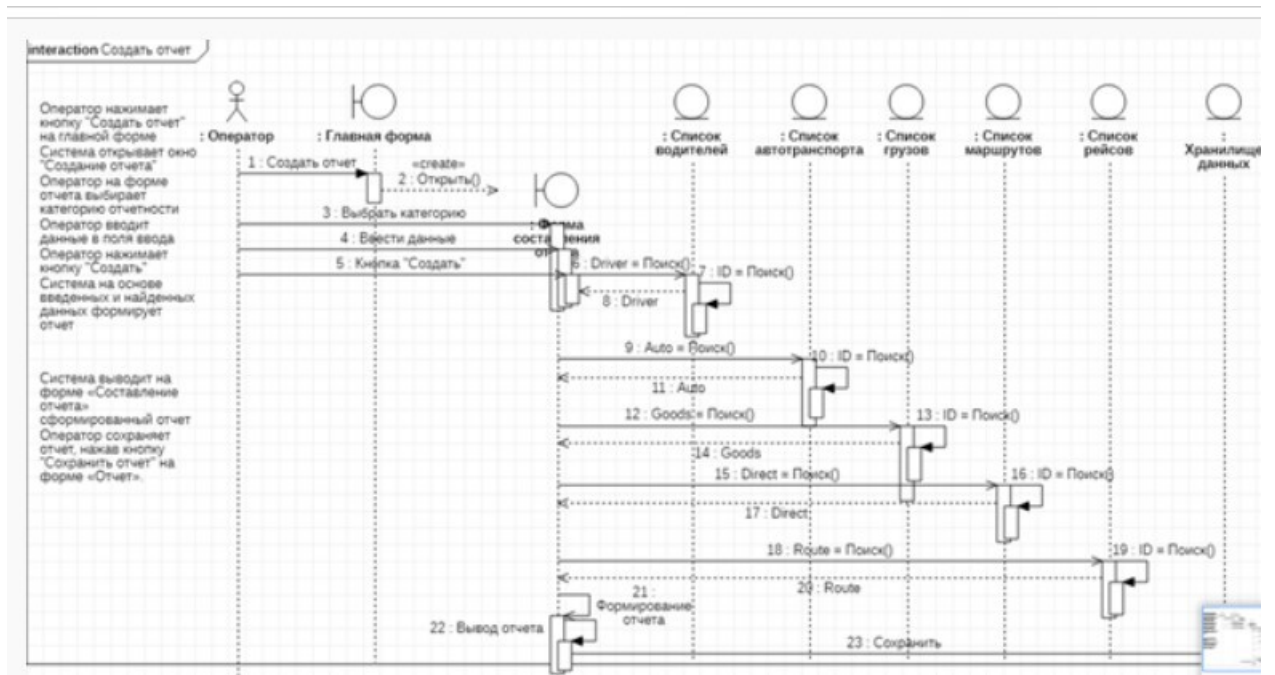


Рисунок 4 - Диаграмма последовательности

3 Разработка инфологической схемы базы данных

Имеется автопарк грузовых автомобилей, коллектив водителей. В базе должна храниться информация о марке машины, номере, грузоподъемности и техническом состоянии автомобиля. О водителях должны храниться следующие данные: фамилия, имя, отчество, год рождения, год начала работы в данной организации, рейтинг водителя по надежности, числящийся за водителем автомобиль. В поставках хранятся данные о водителе, доставляющем груз, дате отправки груза, категории товара, пункте назначения.

Информацию из БД будет использовать оператор информационной системы для планирования грузоперевозок.

База данных (transportation) будет состоять из шести таблиц:

- водитель (КодВодителя, КодАвтомобиля, Фамилия, Имя, Отчество, Классность, Стаж, Возраст, Контактный телефон);
- автомобиль (КодАвтомобиля, Номер, Марка, Грузоподъемность, Техническое состояние);
- груз (КодГруза, Наименование груза, Требования к транспортировке);
- маршрут (КодМаршрута, Место назначения, Расстояние в км, Сложность маршрута);
- рейс (КодВодителя, КодГруза, КодМаршрута, КодОператора, Время);
- оператор (КодОператора, Фамилия, Имя, Отчество, Логин, Пароль, E-mail).

В Приложении А представлено ТЗ на разработку информационной системы.

4 Программные средства для разработки ИС для предметной области «Предприятие автоперевозок»

Прежде чем приступить к проектированию и разработке информационной системы, применимой к предметной области данного курсового проекта, было принято решения рассмотреть уже существующие решения задачи, поставленной в данной работе.

Следующие программные комплексы специально разработаны для экспедиторов, перевозчиков, диспетчеров и позволяют вести документооборот и бухгалтерский учет в транспортных компаниях.

Программный комплекс «Мегалогист» разработан на платформе «1С:Предприятие 8». Предназначен для комплексной автоматизации транспортной логистики. Программа позволяет создавать задания на перевозку, планировать маршруты в ручном и автоматическом режиме, контролировать выполнение рейсов в онлайн-режиме, проводить анализ KPI и рентабельности доставки.

Программа «Грузоперевозки» позволяет принимать и контролировать заявки на перевозку, хранение, перевалку, загрузку, разгрузку, страхование и т.п. услуги для стандартных, сборных, негабаритных, мультимодальных и др. видов грузов. Есть тарифные сетки, учет ГСМ, складской учет, планирование и контроль ремонтов технических средств, расчет зарплаты, формирование всех видов документации, а также различных видов отчетов, анализ и контроль долгов, расчет рентабельности по транспорту, менеджерам и всему предприятию в целом. Отправка e-mail, смс, встроенная проверка контрагентов по множеству параметров в системе «Контур.фокус».

Программа для автоматизации контроля, учета и управления на автотранспортных предприятиях и транспортных подразделениях. Рентабельность от 10 единиц автотранспорта/спецтехники. Полный контроль транспорта, ГСМ, заявок, персонала, ТМЦ, сроков, процессов. Автоматизация управления, интеграция GPS/Глонасс, умный контроль данных, экономия до

20% всех затрат предприятия, резкое снижение ручной работы до 40%, исключение ошибок персонала и фальсификаций данных.

Программа «автоперевозки 4» ведет учет ГСМ и путевых листов, хранит информацию о водителях, автотранспорте, клиентах, показания спидометра, время работы водителя, рассчитывает многие показатели и формирует уже готовый отчет.

Онлайн сервис позволяет автоматизировать все процессы в транспортном предприятии. С помощью данного сервиса вы легко сможете создавать и контролировать заявки, вести учет актов и ТТН, формировать зарплату и отчеты.

В ходе анализа систем аналогов можно сделать вывод, что системы обладают огромной мощностью обработки информации и широчайшими возможностями. В связи с этим стоимость и поддержка данных систем обходится очень дорого. Также люди, которые будут работать с данными системами должны проходить специальное обучение, а так как развитие систем не стоит на месте, то обучение будет многократным и периодическим, что несет за собой дополнительные финансовые траты.

При анализе пяти вышеприведенных работ, было выяснено, что на рынке представлен широкий спектр программ, предоставляющих различные возможности по планированию и автоматизации деятельности транспортной организации. На основании сделанного вывода было принято решение, что проектируемая информационная система будет обладать узконаправленным функционалом под конкретное транспортное предприятие, обладать мощностью и возможностями для решения определенного спектра задач.

Для разработки информационной системы было принято использовать IDE Microsoft Visual Studio, подход к проектированию Code-First. Это обусловлено тем, что данная среда разработки предоставляет удобный и зарекомендовавший себя набор инструментов для создания комфортного интерфейса пользователя. А также фреймворк Entity Framework, представляющий специальную объектно-ориентированную

технологиию на базе фреймворка .NET для работы с данными. Entity Framework представляет собой высокий уровень абстракции, который позволяет абстрагироваться от самой базы данных и работать с данными независимо от типа хранилища. Один из способов взаимодействия с базой данных, предложенных технологией Entity Framework, - Code First. При данном методе разработчик создает класс модели данных, которые будут храниться в базе данных, а затем Entity Framework по этой модели генерирует базу данных и ее таблицы.

Был сделан вывод, что благодаря наличию функционала для разработки удобного интерфейса пользователя, инструментов для работы с базой данных, а также бесплатной версии программы, - IDE Microsoft Visual Studio 2019 предлагает все необходимые функции для решения задачи по разработке информационной системы.

5 Проектирование интерфейса приложения

Пользовательский интерфейс – это совокупность программных и аппаратных средств, обеспечивающих взаимодействие пользователя с компьютером. Основу такого взаимодействия составляют диалоги. Под диалогом в данном случае понимают регламентированный обмен информацией между человеком и компьютером, осуществляемый в реальном масштабе времени и направленный на совместное решение конкретной задачи. Каждый диалог состоит из отдельных процессов ввода/вывода, которые физически обеспечивают связь пользователя и компьютера. Обмен информацией осуществляется передачей сообщения.

Интерфейс планируется максимально простым в использовании, не требующим от оператора какой-либо специальной подготовки или дополнительного обучения. Одна кнопка на главной форме для входа в приложение с названием «Авторизация»

Для добавления данных предусмотрена кнопка с соответствующим названием «Добавить», при нажатии на которую появится возможность выбрать категории данных для добавления – кнопки «Водители», «Автотранспорт», «Грузы», «Маршруты», «Рейсы».

Для составления отчета предусмотрена кнопка «Составить отчет».

Особое внимание в проектировании системы уделено простоте в восприятии пользователем приложения: минимум кнопок – максимум функционала.

На главной форме до прохождения процедуры авторизации все кнопки, кроме «Авторизация», не активны. Главная форма представлена на рисунке 5.

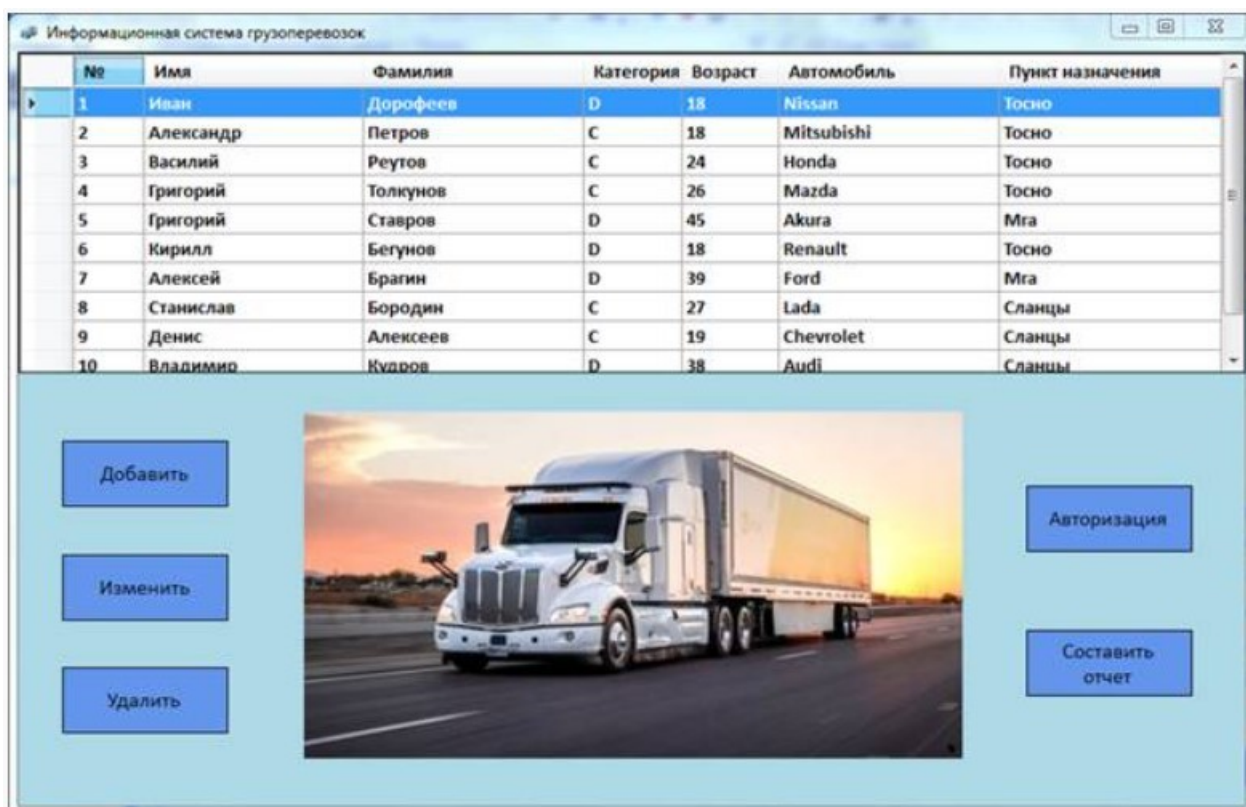


Рисунок 5 - Главная форма приложения

При нажатии кнопки добавить оператор выбирает категорию добавления данных. На рисунке 6 приведен пример добавления данных нового водителя.

Данные о водителе

Водитель

Имя: Иван

Фамилия: Иванов

Отчество: _____

Возраст: 18

Классность: _____

Контактный телефон: _____

Стаж: _____

Подтвердить Отменить

Рисунок 6 - Добавление данных водителя

Аналогичным образом происходит редактирование и удаление данных из нужной категории: оператор выбирает действие – «Изменить» или «Удалить», затем категорию, далее конкретные данные для модификации.

При нажатии кнопки «Составить отчет» на открывшейся форме «Создание отчета» находятся кнопки выбора типа отчета, представленные на рисунке 7.

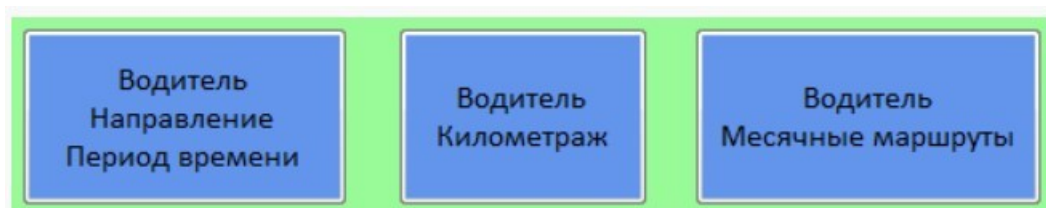


Рисунок 7 - Кнопки выбора типа отчета

После выбора оператором типа отчета, сформированный отчет выводится на форму «Создание отчета».

Как продемонстрировано выше, интерфейс приложения соответствует поставленным требованиям: минимален и интуитивно понятен.

6 Программа и методика испытаний информационной системы

Целью проведения испытаний является проверка на работоспособность информационной системы и определение полноты и качества выполнения функциональных требований.

Проверяемые функции:

- авторизация пользователя;
- ввод данных пользователем.

Тест Кейс 1

Название:

Авторизация оператора в системе

Действие

Ожидаемый результат

Результат теста:

пройден

провален

заблокирован

Предусловие:

Запустите приложение на компьютере

Приложение запущено, открыта главная форма.

пройден

Шаги теста:

Нажмите кнопку «Авторизация»

Заполните поля для авторизации:

"Логин": Оператор

"Пароль": Логист

Открыта форма «Авторизация»

пройден

Нажмите кнопку "Подтвердить"

Кнопки главной формы разблокированы
пройден
Постусловие:
Кликните на кнопке «Создать отчет»
Открыта форма «Создание отчета»
пройден
Тест Кейс 2
Название:
Ввод новых данных оператором
Действие
Ожидаемый результат
Результат теста:
пройден
провален
заблокирован
Предусловие:
Нажата кнопка «Добавить»
Открыта форма «Данные водителя»
пройден
Шаги теста:
Заполните поля для ввода:
Имя: Иван
Фамилия: Иванов
Отчество: Иванович
Возраст: 20
Классность: А
Контактный телефон: 8(900)555-55-55
Стаж: 2
Данные успешно введены

пройден

Нажмите кнопку "Подтвердить"

Поля для ввода очищены

пройден

Постусловие:

Кликните по «Назад»

Главная форма открыта

пройден

7 Создание динамической Web-страницы. Понятие объекта, функции, метода и события. Web-службы. Понятие динамической Web-страницы

Времена, когда веб-сайты состояли из статического контента и нескольких cgi или java скриптов для обработки форм, ушли, безвозвратно. Сейчас от веб-приложений требуется гораздо больше интерактивности. Количество разнообразных технологий и подходов для веб-разработок в настоящее время кажется просто гигантским, а направление их развития - слабопредсказуемым. Как выбрать перспективную рабочую платформу для веб-разработок, действительно удовлетворяющую современным требованиям? Для такого выбора необходимо сравнивать множество подходов и реализаций, нужно иметь о них представление.

Первоначально перед Web-серверами стояла простая задача: найти и отправить клиенту файл, указанный в полученном от клиента запросе. Запрос составлялся тоже очень просто в адресе URL.

Прикладное программирование для Web начиналось с обработки запросов пользователя, передаваемых через формы и динамической генерации страниц на стороне сервера. По этому же принципу работают языки программирования вставок (SSI) в HTML документы. Дальнейшее развитие привело к созданию языков программирования элементов HTML документов на стороне клиента (например, Java Script).

Потом понадобилось сделать на сервере какую-либо небольшую предварительную обработку отправляемого файла. Web-сервер усложнился, появились различные приемы динамической генерации страниц HTML. Затем возникла необходимость выполнять на сервере процедуры. В запросе URL вставили вызов процедур, а на сервере реализовали технологию CGI (Common Gateway Interface). Теперь в запросе URL указывается процедура, которую нужно выполнить на сервере. Процедуру CGI можно написать на любом языке, лишь бы он воспринимал стандартный ввод и стандартный вывод. В технологии Java для этого создаются апплеты, сервлеты, используется язык

JSP(Java Server Pages).

Причины возрастания роли веб- приложений понятны - они не требуют установки программных средств у пользователя и их гораздо проще "подстраивать под этого самого пользователя", такие приложения более управляемы с обеих сторон, меньше требований к клиентскому устройству. Многие приложения уже используют для взаимодействия с пользователем веб - интерфейс. То есть в веб-приложения закладывается функциональность, сравнимую с традиционными настольными приложениями.

Преимущества приложений даны в табл. 1.

В основе реализации корпоративных информационных систем на базе архитектуры Интернет/Интранет лежит принцип "открытой архитектуры", что во многом определяет независимость реализации корпоративной системы от конкретного производителя. Все программное обеспечение таких систем реализуется в виде апплетов или сервлетов (программ написанных на языке JAVA) или в виде cgi модулей (программ написанных, как правило, на Perl или C).

Под клиентской платформой целесообразно понимать не только системное окружение на клиентской стороне, но и способ организации пользовательского интерфейса и его взаимодействия с бизнес-логикой, разделенной в рамках приложения на клиентскую и серверную часть. В приложении осуществляется взаимодействие между клиентской и серверной частью, и является определяющим для клиентской платформы.

При обеспечении WEB-доступа к существующим БД, возможен ряд технологических и организационных решений. Практика использования WEB - технологии для доступа к существующим БД предоставляет широкий спектр технологических решений, по разному связанных между собой - перекрывающихся, взаимодействующих и дополняющих. Выбор конкретных решений при обеспечении доступа зависит от специфики конкретной СУБД и от ряда других факторов, как платформа, сервер, наличие специалистов, способных с минимальными издержками освоить определенную ветвь

технологических решений, существование других БД.

Таблица 1 - Преимущества динамических приложений

Преимущество	Метод реализации
Централизация администрирования приложений	За счет централизованного оперирования приложениями и их модификациями, выполняемыми на сервере, они становятся доступными для всех пользователей сразу, не требуется контакт с отдельными пользователями
Упрощение технологии обслуживания рабочих мест	Применяя соответствующие сервисные средства, администратор системы может одновременно обслуживать множество устройств.
Возможность контроля за действиями пользователя	Благодаря отсутствию накопителей на рабочем месте, пользователь не может привносить в конфигурацию программного обеспечения что-то свое, устанавливая собственные программы.
Мобильность пользователей	Пользователь, не привязанный к конкретному рабочему месту, может произвольно перемещаться в пределах сети, применять устройства дистанционного доступа.
Повышение производительности труда операторов	Сведение всех сервисных операций на сервер заметно повышает производительность труда операторов.
Снижение стоимости эксплуатации оборудования	При не слишком большом различии в стоимости оборудования серверные приложения заметно дешевле в эксплуатации.
	Высокая способность к интеграции существующих гетерогенных информационных ресурсов.
	Минимальный состав программно-технических средств на клиентском рабочем месте (Браузер и общесистемное программное обеспечение)
	Минимальные затраты на настройку и сопровождение клиентских рабочих мест, что позволяет реализовывать системы с тысячами пользователей (причем многие из которых могут работать за удаленными терминалами).

В общем случае информационная система, реализованная с использованием данной архитектуры, включает Web-узлы с интерактивным информационным наполнением, реализованных при помощи технологий Java, JavaBeans, JavaScript, PHP, ASP, Perl, взаимодействующих с базой данных, с

одной стороны, и с клиентским местом с другой. База данных, в свою очередь, является источником информации для интерактивных приложений реального времени.

Динамические веб-страницы содержат программный код на специальных языках веб-программирования. Практически все крупные сайты состоят из динамических веб-страниц.

Процедура передачи динамических веб-страниц по запросу пользователя состоит из большого количества действий, реализованных с участием языков веб-программирования.

Динамическими также называют веб-страницы, которые хранятся на сервере в готовом виде, но содержат программный код. Этим обеспечивается интерактивность веб-страницы, т. е. такая веб-страница «реагирует» на действия пользователя.

Языки веб-программирования можно условно разделить на две пересекающиеся группы: клиентские и серверные.

Серверные языки могут взаимодействовать с сервером. Программный код клиентских языков выполняет браузер на компьютере-клиенте, поэтому сервер не загружается дополнительной работой.

Для создания динамических страниц с программным кодом на клиентских языках используют технологию, получившую название Dynamic HTML (DHTML).

Эта технология представляет собой комбинацию из языка разметки гипертекста (HTML), клиентского языка сценариев (JavaScript) и языка описания стилей (CSS).

Программа на языке JavaScript называется сценарием или скриптом. С помощью скрипта можно изменять содержимое и оформление веб-страницы в ответ на действия пользователя:

- заменять текст, оформление, рисунки;
- строить многоуровневые выпадающие меню;

8 Способы разработки Web-страниц. Языки гипертекстовой разметки

Существует несколько способов, с помощью которых можно создать web-страницу.

Первый способ — с помощью онлайн редактора. Онлайн редакторы — это сайты, которые создаются в интернете на хостинге (Хостинг — место, где размещается сайт в сети интернет). Также в сети Интернет можно без труда найти бесплатные шаблоны, с помощью которых можно сделать себе сайт.

Второй способ — оффлайн редакторы — программы, которые устанавливаются на компьютер. В отличие от онлайн редакторов, оффлайн редакторы привязаны к одному компьютеру, который даже может быть не привязан к интернету, что в случае с первым способом невозможно. Основной плюс таких редакторов в том, что они гораздо мощнее, на них можно создавать очень сложные и универсальные проекты, не имея подключения к интернету.

Существуют и более легкие удобные редакторы, которые будут полезны для новичков — так называемые визуальные редакторы. С их помощью можно быстро создавать HTML страницы, даже не зная языка программирования.

SGML (англ. Standard Generalized Markup Language — стандартный обобщённый язык разметки; произносится [эс-джи-эм-эл]) — метаязык, на котором можно определять язык разметки для документов. SGML — наследник разработанного в 1969 году в IBM языка GML (Generalized Markup Language), который не стоит путать с Geography Markup Language, разрабатываемым Open GIS Consortium.

Изначально SGML был разработан для совместного использования машинно-читаемых документов в больших правительственных и аэрокосмических проектах. Он широко использовался в печатной и издательской сфере, но его сложность затруднила его широкое распространение для повседневного использования.

XML - Extensible Markup Language, то есть Расширяемый Язык Разметки, возник в результате развития языка HTML (HyperText Markup Language, языка

разметки гипертекста). Однако было бы грубой ошибкой воспринимать его лишь как некую усовершенствованную версию языка HTML. По сути, язык XML представляет собой новое поколение языков разметки. Здесь следует иметь в виду 3 момента, принципиально отличающих XML от HTML и его предшественников (GML - Generalized Markup Language, SGML - Standard General Markup Language):

XML, в отличие от HTML, НЕ ИМЕЕТ ПРЕДОПРЕДЕЛЕННЫХ ТЭГОВ - точнее, каждый разработчик может создавать СВОИ СОБСТВЕННЫЕ XML-тэги - столько, сколько нужно. Количество таких тэгов практически неограничено. Таким образом, XML является метаязыком, позволяющим создавать другие языки разметки, такие как, например, HTML.

Hypertext Markup Language (HTML) это простая система для создания гипертекстовых документов, которые переносимы с одной платформы на другую. В сущности, HTML документы это SGML документы с общей семантикой, которая подходит для представления информации в большом количестве приложений. HTML может использоваться для представления:

- Гипертекстовых новостей, почты, документации и гипермедиа
- Меню опций
- Результатов запросов к базам данных
- Простых документов с встроенными графическими образами
- А также, для гипертекстового просмотра существующих массивов информации

Проект World Wide Web (WWW) объединяет разбросанную по миру информацию. Для этого WWW использует Internet HyperText Transfer Protocol (HTTP) - Протокол Передачи Гипертекста, который позволяет ввести соглашения на способы представления данных при взаимодействии клиента и сервера.

Данные передаются в теле сообщения в соответствии с MIME стандартом. HTML это одно из представлений информации в WWW.

Заключение

В работе проанализированы особенности предметной области и объекта автоматизации. Обоснован выбор необходимости разработки ИС с помощью объектно-ориентированной методологии. Приведены характеристики диаграмм UML.

Разработаны модель функциональных требований к системе в виде диаграммы вариантов использования, модель поведения системы в виде диаграмм действий, модель логической структуры системы в виде диаграмм классов, модель взаимодействия объектов системы в виде диаграмм последовательности.

Внедрение данной задачи в эксплуатацию даст возможность облегчить учет заказов на грузоперевозку, путем освобождения пользователя от работы с бумажными документами. Доступ, хранение, и редактирование данных станет намного проще и удобнее. Ситуация переделывания каких-либо документов упростится так как не нужно будет переделывать все целиком заново, а потребуется только редактировать уже существующую информацию.

Проектируя информационную систему, мы познакомились IDE Microsoft Visual Studio 2019. С её помощью можно быстро создавать деловые приложения для различных сфер деятельности человека. Удобный интерфейс и широкий функционал. Единственным относительным минусом при работе с данной средой оказался довольно высокий порог вхождения для неопытного пользователя.

Разработана информационная система учета работы компании по автоперевозкам.

Разработаны инструкции по запуску программы и руководство пользователя.

Информационная система может быть дополнена дополнительным функционалом.

Список использованной литературы

1. Савин В.И. Перевозки грузов автомобильным транспортом: Справочное пособие. – М.: Издательство «Дело и Сервис», 2018 – 544 с.
2. Единая транспортная система: учебник для студ. Учреждений сред. Проф. Образования. – 9-е изд., стер. – М.: Издательский центр «Академия», 2017. – 240 с.
3. Новые направления в мониторинге [Электронный ресурс]. - Режим доступа: <http://vestnik-glonass.ru/stati/novye-napravleniya-vmonitoringe->.
4. Варфоломеева, А.О. Информационные системы предприятия: Учебное пособие. – М.: НИЦ ИНФРА-М, 2019. – 283 с.
5. Голицина, О.Л., Максимов Н.В., Попов И.И. Базы данных: Учебное пособие. – М.: Формум: ИНФРА-М, 2018. – 352 с.
6. Заботина, Н.Н. Проектирование информационных систем – М.: ДРОФА, 2018. – 336 с.
7. Мезенцев, К.Н. Автоматизированные информационные системы: учебник для студентов учреждений среднего профессионального образования. – М.: ИЦ Академия, 2018. – 176 с.

Приложение А

Техническое задание на разработку информационной системы

Техническое задание

На разработку информационной системы учета работы автотранспортной компании

1 Введение

Полное наименование системы: «Информационная системы учета работы автотранспортной компании «СПб-ЛО Перевозки»».

Система предназначена для применения в автотранспортной компании, занимающаяся перевозкой различных грузов.

2 Назначение и цели создания информационной системы

2.1 Назначение информационной системы

Информационная система разрабатывается с целью оптимизации использования ресурсов компании: эффективного отслеживания деятельности организации по грузоперевозкам, составления отчетов в логистический отдел компании, которые помогут эффективно распределять по необходимым направлениям (маршрутам) грузы и автотранспорт, в зависимости от сложности маршрута, назначать в рейс водителей с соответствующим опытом и навыками. Для повышения эффективности перевозок в целом.

2.2 Цели создания информационной системы

Информационная система создается с целью:

- сбора исходной информации, необходимой для планирования грузоперевозок и подготовки отчетности по показателям деятельности;
- повышения качества (полноты, точности, достоверности, своевременности, согласованности) информации;
- повышение эффективности использования водителей в зависимости от стажа и опыта;
- увеличение количества возможных перевозок с помощью аналитики;

- эффективное распределения грузов по соответствующему автотранспорту;

- сокращение времени, затрачиваемого на информационно-аналитическую деятельность;

3 Функциональные задачи информационной системы

3.1 Ввод, редактирование, удаление данных

Оператор осуществляет ввод, удаление и редактирование той или иной информации, с возможностью автоматизации (подстановок из других таблиц, фиксированных наборов данных и т.д.) и входного контроля от некорректных действий. Оператор вводит (редактирует, удаляет) данные в следующие категории:

- водитель;
- автотранспорт;
- груз;
- направление (маршрут);
- рейс.

3.2 Идентификация пользователя

Ограничение доступа к информационной системе с помощью идентификации пользователя (парольная защита).

3.3 Поиск данных

Оператор осуществляет поиск по разнообразным признакам (не менее одного в каждой из информационных категорий, например, ФИО водителя и т.д.). С помощью функции поиска можно получить информацию об отдельной единице категории или совокупности объектов, занесённых в информационную систему.

3.4 Сортировка данных

Оператор осуществляет сортировку по различным признакам (не менее одного в каждой из информационных категорий, например, по характеристикам перевозимых грузов и т.д.).

3.5 Основные запросы для составления отчетов

Оператор, заполняя соответствующие поля для запроса в системе, имеет возможность быстро находить соответствующую информацию для формирования отчёта по основным логистическим вопросам компании.

Основные запросы представляют собой три совокупности данных:

3.5.1 Водитель – направление – период времени

Информация по всем водителям, которые совершали перевозки грузов за выбранный промежуток времени в заданном направлении.

3.5.2 Водитель – километраж

Информация о том, кто за заданный промежуток времени совершил максимальные (в километровом исчислении) перевозки.

3.5.3 Водитель – месячные маршруты

Информация для любого заданного водителя, о всех маршрутах, в которые он ездил в текущем месяце.

4 Требования к системе

4.1 Технические требования к системе

4.1.1 Основные требования к функционалу

На главной форме должны быть предусмотрены кнопки для быстрого доступа к вызову форм ввода, редактирования и удаления данных по основным категориям:

- водители;
- автомобили;
- грузы;
- маршруты;
- рейсы.

А также кнопка для перехода в окно составления отчета с главной формы.

4.1.2 Требования к ПК и программному обеспечению

Операционная система: Windows XP/Vista/7/8.1/10

Процессор: Intel Pentium D 3 GHz и более современные аналоги

Оперативная память: 2 Gb

4.2 Входные данные

Входными данными являются данные, вводимые пользователем с клавиатуры:

- данные водителя: ФИО, дата рождения, классность, стаж, контактный телефон;
- сведения об автомобиле: марка, номер, грузоподъемность и мощность, техническое состояние;
- сведения о характеристиках груза: тип груза, требования к транспортировке;
- сведения о маршрутах: место назначения, расстояние в километрах, класс сложности маршрута;
- сведения о текущих рейсах: маршрут, кто и на какой машине повезет какой-либо груз, за какое время.

4.3 Обработка данных

- при помощи кнопки «Добавить» значения, введенные оператором, записываются в таблицу.
- при помощи кнопки «Изменить» значения в таблице изменяются.
- предназначение кнопки «Удалить» - удаление выбранной строки из базы данных.
- при помощи соответствующих полей оператор вводит нужные данные для поиска необходимой информации. При нажатии кнопки «Поиск» производится поиск данных.

4.4 Выходные данные

Выходными данными для продукта являются значения, выводимые программой пользователю на экран в процессе обработки данных.

4.5 Требования к лингвистическому обеспечению

Система должна поддерживать следующие языки графического интерфейса:

- русский.

4.6 Внешний вид приложения

Использование приложения должно быть комфортным для

неподготовленного пользователя, не требующим дополнительного обучения по эксплуатации.

4.7 Профиль пользователя

Должна быть предусмотрена аутентификация пользователя, соответственно наличия у пользователя логина и пароля, а также информации о личных данных:

- имя;
- фамилия;
- дата рождения;
- должность.

5 Разработка интерфейса системы

Вид главной формы с доступом к редактированию данных, составлению отчета, авторизации представлен на рисунке 8.



Рисунок 8 - Главная форма

Вид формы ввода и редактирования данных водителя представлен на

рисунке 9.

Рисунок 9 - Форма ввода и редактирования данных водителя

Вид формы ввода и редактирования данных автомобиля представлен на рисунке 10.

Рисунок 10 - Форма ввода и редактирования данных водителя

6 Пользователи системы

Система предназначена для должности «Оператор ЭВМ», предусматривающей работу с различными приложениями. Возможна работа больше одного человека за одним компьютером, поэтому предусматривается парольная защита для доступа к информационной системе.

Приложение Б

Разработка технической документации на программный продукт

Разработка руководства пользователя

1 Введение

1.1 Область применения

Пользовательский интерфейс ИС УРТК обеспечивает информационную поддержку деятельности оператора ИС УРТК при выполнении следующих операций: Регистрации сведений о перевозках, Формирование отчетности для логистического отдела.

1.2 Краткое описание возможностей

ИС УРТК обеспечивает выполнение следующих основных функций:

- регистрация пользователей ИС УРТК;
- ведение перечня перевозок;
- формирование отчетности для логистического отдела.

1.3 Уровень подготовки пользователя

Для эксплуатации ИС УРТК определены следующие роли:

- Оператор.

Оператор должен:

- Иметь общие сведения о системе и ее назначении;
- Владеть информацией об ИС УРТК в объеме эксплуатационной документации;
- Владеть информацией о работе в интерфейсе ИС УРТК;
- Осуществлять ведение справочников в ИС УРТК;
- Обеспечивать поддержку взаимодействия с внешними участниками ИС УРТК (обмен данными, регистрация обращений);
- Формировать аналитическую отчетность в ИС УРТК.

1.4 Перечень эксплуатационной документации

1 Инструкция по установке ИС УРТК;

- 2 Руководство администратора ИС УРТК;
- 3 Руководство по техническому обслуживанию ИС УРТК;
- 4 Руководство пользователя ИС УРТК (настоящий документ).

1.5 Список обозначений и сокращений

БД - База данных

АРМ - Автоматизированное рабочее место

ИС - Информационная Система

1.6 Назначение и условия применения

1.6.1 Назначение

АРМ УРТК предназначен для осуществления информационной поддержки ведения реестра грузоперевозок, а также формирования различных отчетов для отдела логистики.

АРМ УРТК позволяет:

- обеспечить взаимодействие с информационными системами поставщиков услуг;

- повысить прозрачность деятельности в сфере оказания услуг заказчикам за счет формирования полной, достоверной и актуальной информации о ходе и результатах проведения грузоперевозок;

- повысить эффективность контроля и анализа процесса грузоперевозок.

1.6.2 Условия применения

АРМ УРТК может эксплуатироваться и выполнять заданные функции при соблюдении требований, предъявляемых к техническому, системному и прикладному программному обеспечению.

1.6.3 Описание операций

В данном разделе приводится описание всех операций, существующих в АРМ УРТК.

Роль: Оператор

Доступные операции: Просмотр, ввод, редактирование, сортировка, удаление данных; формирование отчетов.

1.6.3.1 Вход в АРМ УРТК

Для входа в пользовательский интерфейс АРМ УРТК необходимо запустить приложение с помощью ярлыка на рабочем столе или соответствующего пункта в меню «Пуск».

Для получения прав доступа к АРМ необходимо предварительно обратиться к администратору УРТК. Для доступа к данным необходимо выполнить авторизацию: указать логин и пароль, а затем нажать кнопку «Войти». Пользователь имеет 3 попытки для корректного входа в АРМ. Если ошибочный пароль введен более 3 раз, то доступ к АРМ для этого пользователя блокируется и ему необходимо обратиться к администратору.

1.6.3.2 Регистрация пользователей системы

Регистрация пользователей системы доступна пользователю с ролью Администратор. Администратор имеет право назначать пользователям роли: Администратор и Оператор. Оператору список пользователей доступен для просмотра. Для создания записи о пользователе модуля необходимо войти в пункт меню «Регистрация пользователей» и нажать кнопку «Добавить».

В открывшемся окне необходимо заполнить поля:

- Логин;
- Пароль;
- Фамилия;
- Имя;
- Отчество;
- Номер телефона;
- Должность;

Необходимо использовать пароли не менее 8 символов. Пароль должен содержать хотя бы одну цифру и заглавную букву. Для сохранения данных необходимо нажать кнопку «Ок». Если сохранение данных не требуется, то следует нажать кнопку «Отмена».

1.6.3.3 Добавление сведений о перевозках

Добавление сведений о перевозках доступно пользователю с ролью

Оператор. Для создания записи необходимо войти в пункт меню «Добавить» и нажать кнопку «ОК». Будет предложено выбрать пять категорий для добавления данных:

- водители;
- автотранспорт;
- грузы;
- маршруты;
- рейсы.

В каждой из категории нужно заполнить экранные формы.

Водители:

- фамилия;
- имя;
- отчество;
- дата рождения;
- классность;
- стаж;
- контактный телефон.

Автотранспорт:

- марка;
- номер;
- грузоподъемность;
- техническое состояние.

Грузы:

- тип груза;
- требования к транспортировке.

Маршруты:

- место назначения;
- расстояние в километрах;
- класс сложности маршрута.

Рейсы:

- маршрут;
- водитель;
- автомобиль;
- груз;
- время.

1.7 Формирование отчета

Для формирования отчетов следует в АРМ УРТК выбрать пункт «Отчеты» главного меню. При этом пользователю предоставляется возможность из выпадающего меню выбрать пункт для формирования одного из следующих отчетов:

Маршруты за период времени

Оператор должен ввести данные для формирования отчета:

- водитель;
- направление;
- период времени.

Километраж по перевозкам

Оператор должен ввести данные для формирования отчета:

- водитель;
- период времени.

Отчет по конкретному водителю за месяц

- водитель;
- подотчетный месяц.

Для завершения формирования отчета следует нажать кнопку «Сформировать отчет». В случае успешного выполнения операции загружается экранная форма, содержащая сформированный отчет. В противном случае Системой будет выдано сообщение об ошибке.

Приложение А - Техническое задание на разработку

Техническое задание

На разработку информационной системы учета работы автотранспортной компании

1 Введение

Полное наименование системы: «Информационная системы учета работы автотранспортной компании «СПб-ЛО Перевозки»».

Система предназначена для применения в автотранспортной компании, занимающаяся перевозкой различных грузов.

2 Назначение и цели создания информационной системы 2.1

Назначение информационной системы

Информационная система разрабатывается с целью оптимизации использования ресурсов компании: эффективного отслеживания деятельности организации по грузоперевозкам, составления отчетов в логистический отдел компании, которые помогут эффективно распределять по необходимым направлениям (маршрутам) грузы и автотранспорт, в зависимости от сложности маршрута, назначать в рейс водителей с соответствующим опытом и навыками. Для повышения эффективности перевозок в целом.

Цели создния информационной системы

2.2 Цели создания информационной системы

Информационная система создается с целью

- сбора исходной информации, необходимой для планирования грузоперевозок и подготовки отчетности по показателям деятельности;
- повышения качества (полноты, точности, достоверности, своевременности, согласованности) информации;
- повышение эффективности использования водителей в зависимости от стажа и опыта;
- увеличение количества возможных перевозок с помощью аналитики;
- эффективное распределения грузов по соответствующему автотранспорту;
- сокращение времени, затрачиваемого на информационно-аналитическую деятельность;

3 Функциональные задачи информационной системы 3.1 Ввод, редактирование, удаление данных

Оператор осуществляет ввод, удаление и редактирование той или иной информации, с возможностью автоматизации (подстановок из других таблиц, фиксированных наборов данных и т.д.) и входного контроля от некорректных действий. Оператор вводит (редактирует, удаляет) данные в следующие категории:

- водитель;
- автотранспорт;

- груз;
- направление (маршрут);
- рейс.

3.2 Идентификация пользователя

Ограничение доступа к информационной системе с помощью идентификации пользователя (парольная защита).

3.3 Поиск данных

Оператор осуществляет поиск по разнообразным признакам (не менее одного в каждой из информационных категорий, например, ФИО водителя и т.д.). С помощью функции поиска можно получить информацию об отдельной единице категории или совокупности объектов, занесённых в информационную систему.

3.4 Сортировка данных

Оператор осуществляет сортировку по различным признакам (не менее одного в каждой из информационных категорий, например, по характеристикам перевозимых грузов и т.д.).

3.5 Основные запросы для составления отчетов

Оператор, заполняя соответствующие поля для запроса в системе, имеет возможность быстро находить соответствующую информацию для формирования отчёта по основным логистическим вопросам компании.

Основные запросы представляют собой три совокупности данных:

3.5.1 Водитель – направление – период времени

Информация по всем водителям, которые совершали перевозки грузов за выбранный промежуток времени в заданном направлении.

3.5.2 Водитель – километраж

Информация о том, кто за заданный промежуток времени совершил максимальные (в километровом исчислении) перевозки.

3.5.3 Водитель – месячные маршруты

Информация для любого заданного водителя, о всех маршрутах, в которые он ездил в текущем месяце.

4 Требования к системе 4.1 Технические требования к системе 4.1.1

Основные требования к функционалу

На главной форме должны быть предусмотрены кнопки для быстрого доступа к вызову форм ввода, редактирования и удаления данных по основным категориям:

- водители;
- автомобили;
- грузы;
- маршруты;
- рейсы.

А также кнопка для перехода в окно составления отчета с главной формы.

4.1.2 Требования к ПК и программному обеспечению

Операционная система: Windows XP/Vista/7/8.1/10

Процессор: Intel Pentium D 3 GHz и более современные аналоги

Оперативная память: 2 Gb

4.2 Входные данные

Входными данными являются данные, вводимые пользователем с клавиатуры:

- данные водителя: ФИО, дата рождения, классность, стаж, контактный телефон;

- сведения об автомобиле: марка, номер, грузоподъемность и мощность, техническое состояние;

- сведения о характеристиках груза: тип груза, требования к транспортировке;

- сведения о маршрутах: место назначения, расстояние в километрах, класс сложности маршрута;

- сведения о текущих рейсах: маршрут, кто и на какой машине повезет какой-либо груз, за какое время.

4.3 Обработка данных

- при помощи кнопки «Добавить» значения, введённые оператором, записываются в таблицу.

- при помощи кнопки «Изменить» значения в таблице изменяются.

- предназначение кнопки «Удалить» - удаление выбранной строки из базы данных.

- при помощи соответствующих полей оператор вводит нужные данные для поиска необходимой информации. При нажатии кнопки «Поиск» производится поиск данных.

4.4 Выходные данные

Выходными данными для продукта являются значения, выводимые программой пользователю на экран в процессе обработки данных.

4.5 Требования к лингвистическому обеспечению

Система должна поддерживать следующие языки графического интерфейса:

- русский.

4.6 Внешний вид приложения

Использование приложения должно быть комфортным для неподготовленного пользователя, не требующим дополнительного обучения по эксплуатации.

4.7 Профиль пользователя

Должна быть предусмотрена аутентификация пользователя, соответственно наличия у пользователя логина и пароля, а также информации о личных данных:

- имя;

- фамилия;

- дата рождения;

- должность.

5 Разработка интерфейса системы

Вид главной формы с доступом к редактированию данных, составлению отчета, авторизации представлен на рисунке 10.

Рисунок 10 - Главная форма

Вид формы ввода и редактирования данных водителя представлен на рисунке 11.

Рисунок 11 - Форма ввода и редактирования данных водителя

Вид формы ввода и редактирования данных автомобиля представлен на рисунке 12.

Рисунок 12 - Форма ввода и редактирования данных водителя

6 Пользователи системы

Система предназначена для должности «Оператор ЭВМ»,

предусматривающей работу с различными приложениями. Возможна работа больше одного человека за одним компьютером, поэтому предусматривается парольная защита для доступа к информационной системе.

Приложение Б - Исходный код

Листинг 1 Главная форма:

```
using System;
```

```
using System.Collections.Generic;
```

```
using System.ComponentModel;
```

```
using System.Data;
```

```
using System.Drawing;
```

```
using System.Linq;
```

```
using System.Text;
```

```
using System.Threading.Tasks;
```

```
using System.Windows.Forms;
```

```
using System.Data.Entity;
```

```
namespace TruckingCompany
```

```
{  
  
public partial class Form1 : Form  
  
{  
  
    DriverContext db;  
  
    public Form1()  
  
    {  
  
        db = new DriverContext();  
  
        db.Drivers.Load();  
  
        InitializeComponent();  
  
        dataGridView1.DataSource = db.Drivers.Local.ToBindingList();  
  
    }  
  
    // add  
  
    private void Button1_Click(object sender, EventArgs e)  
  
    {  
  
        DriverForm drvForm = new DriverForm();
```

```
DialogResult result = drvForm.ShowDialog(this);

if (result == DialogResult.Cancel)

    return;

Driver driver = new Driver();

driver.Name = drvForm.textBox1.Text;

driver.Surname = drvForm.textBox2.Text;

driver.Category = drvForm.comboBox1.SelectedItem.ToString();

driver.Age = (int)drvForm.numericUpDown1.Value;

driver.Truck = (Truck)drvForm.comboBox2.SelectedItem;

driver.Point = (Point)drvForm.comboBox3.SelectedItem;

db.Drivers.Add(driver);

db.SaveChanges();

dataGridView1.Refresh();

MessageBox.Show("Новый водитель добавлен");

}
```

```

// update

private void Button2_Click(object sender, EventArgs e)

{

    if (dataGridView1.SelectedRows.Count > 0)

    {

        int index = dataGridView1.SelectedRows[0].Index;

        int id = 0;

                                bool converted = Int32.TryParse(dataGridView1[0,
index].Value.ToString(), out id);

        if (converted == false)

            return;

        Driver driver = db.Drivers.Find(id);

        DriverForm drvForm = new DriverForm();

        drvForm.textBox1.Text = driver.Name;

        drvForm.textBox2.Text = driver.Surname;

        drvForm.comboBox1.SelectedItem = driver.Category;

```

```
drvForm.numericUpDown1.Value = driver.Age;

drvForm.comboBox2.SelectedItem = driver.Truck;

drvForm.comboBox3.SelectedItem = driver.Point;

List trucks = db.Trucks.ToList();

drvForm.comboBox2.DataSource = trucks;

drvForm.comboBox2.ValueMember = "Id";

drvForm.comboBox2.DisplayMember = "Title";

List points = db.Points.ToList();

drvForm.comboBox3.DataSource = points;

drvForm.comboBox3.ValueMember = "Id";

drvForm.comboBox3.DisplayMember = "City";

if (driver.Truck != null)

    drvForm.comboBox2.SelectedValue = driver.Truck.Id;

DialogResult result = drvForm.ShowDialog(this);

if (result == DialogResult.Cancel)
```

```
        return;

        driver.Name = drvForm.textBox1.Text;

        driver.Surname = drvForm.textBox2.Text;

        driver.Category = drvForm.comboBox1.SelectedItem.ToString();

        driver.Age = (int)drvForm.numericUpDown1.Value;

        driver.Truck = (Truck)drvForm.comboBox2.SelectedItem;

        driver.Point = (Point)drvForm.comboBox3.SelectedItem;

        db.Entry(driver).State = EntityState.Modified;

        db.SaveChanges();

        dataGridView1.Refresh();

        MessageBox.Show("Данные обновлены");

    }

}

// delete

private void Button3_Click(object sender, EventArgs e)
```

```
{  
  
    if (dataGridView1.SelectedRows.Count > 0)  
  
        {  
  
            int index = dataGridView1.SelectedRows[0].Index;  
  
            int id = 0;  
  
                bool converted = Int32.TryParse(dataGridView1[0,  
index].Value.ToString(), out id);  
  
            if (converted == false)  
  
                return;  
  
            Driver driver = db.Drivers.Find(id);  
  
            db.Drivers.Remove(driver);  
  
            db.SaveChanges();  
  
            dataGridView1.Refresh();  
  
            MessageBox.Show("Данные удалены");  
  
        }  
}
```



```
}

// open AuthorizationForm

private void Button4_Click(object sender, EventArgs e)

{

    AuthorizationForm AuthF = new AuthorizationForm();

    AuthF.Show();

}

// open ReportForm

private void Button5_Click(object sender, EventArgs e)

{

    ReportForm RepF = new ReportForm();

    RepF.Show();

}

}

}
```