Министерство образования и молодежной политики Свердловской области

государственное автономное профессиональное

образовательное учреждение Свердловской области

«Уральский радиотехнический колледж им. А.С. Попова»

|  |  |
| --- | --- |
|  |  |

Проектирование и разработка программного обеспечения для логистической компании «Быстрее света»

Пояснительная записка к курсовому проекту

по ПМ.03 Участие в интеграции программных модулей

по МДК 03.01 Технология разработки программного обеспечения

РК 09.02.03.332 24 ПЗ

|  |  |  |
| --- | --- | --- |
| Нормоконтролер  \_\_\_\_\_\_\_\_\_\_\_\_ А. В. Беляева  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_2023 г. |  | Руководитель  \_\_\_\_\_\_\_\_\_\_\_\_\_ А. В. Беляева  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2023 г. |
|  |  | Разработчик  \_\_\_\_\_\_\_\_\_\_\_\_\_С. Ю. Хромцов  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2023 г. |
|  |  |  |

Екатеринбург

2023

УТВЕРЖДЕНО

цикловой методической комиссией

информационных технологий

от «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_20\_\_г.

«\_ »\_ Н.Н Шутова

**ЗАДАНИЕ**

для курсового проектирования по ПМ.03 Участие в интеграции программных модулей по МДК 03.01 Технология разработки программного обеспечения студенту(ке) «4» курса группы «\_\_\_\_\_\_\_\_»

ГАПОУ СО «Уральский радиотехнический колледж им А.С. Попова»

Тема задания

Курсовой проект на указанную тему выполняется студентом в следующем объеме:

* 1. Пояснительная записка состоит из следующих разделов:

Введение

1. Постановка задачи
2. Системный проект
   1. Описание предметной области
   2. Диаграммы вариантов использования
   3. Сценарии вариантов использования
   4. Описание данных
   5. Логическая структура базы данных

3 Технический проект

* 1. Выбор состава технических и программных средств
  2. Физическая структура базы данных
  3. Физическая структура программы

1. Экспериментальная часть
   1. Тестирование приложения
   2. Руководство пользователя.

Заключение

Список использованных источников

Дата выдачи « » 20 г

Срок окончания « » 20 г

Преподаватель А.В. Беляева

Содержание

[Введение 4](#_Toc153773377)

[1 Постановка задачи 6](#_Toc153773378)

[2 Системный проект 8](#_Toc153773379)

[2.1 Описание предметной области 8](#_Toc153773380)

[2.2 Диаграммы вариантов использования 10](#_Toc153773381)

[2.4 Описание данных 22](#_Toc153773382)

[2.5 Логическая структура базы данных 27](#_Toc153773383)

[3 Системный проект 30](#_Toc153773384)

[3.1 Выбор состава технических и программных средств 30](#_Toc153773385)

[3.2 Физическая структура базы данных 32](#_Toc153773386)

[3.3 Физическая структура программы 33](#_Toc153773387)

[4 Экспериментальная часть 36](#_Toc153773388)

[4.1 Тестирование приложения 36](#_Toc153773389)

[4.2 Руководство пользователя 42](#_Toc153773390)

[Заключение 56](#_Toc153773391)

[Список использованных источников 57](#_Toc153773392)

Изм.

Лист

№ докум.

Подпись

Дата

Лист

3

РК 09.02.03 332 24 ПЗ

Разраб.

Хромцов С. Ю.

Провер.

Беляева А. В.

Н. Контр.

Беляева А. В.

Проектирование и разработка программного обеспечения для логистической компании «Быстрее света»

Пояснительная записка

Лит.

Листов

58

ГАПОУ CO

УРТК им. А.С. Попова

Введение

Темой данной курсовой работы является проектирование и разработка программного обеспечения для логистической компании «Быстрее света». Так как современный мир, с его высокой степенью глобализации и увеличивающимся объемом товарооборота, ставит перед логистическими компаниями огромные вызовы, в преодолении которых помогает программное обеспечение. В условиях быстро меняющейся бизнес-среды и растущих ожиданий клиентов, эффективное управление логистическими процессами становится ключевым фактором конкурентоспособности. В данном контексте, разработка программного продукта для логистической компании приобретает особую актуальность.

Логистические компании сегодня сталкиваются с рядом сложных проблем. Во-первых, важно учитывать глобальные изменения в торговле, такие как рост электронной коммерции, увеличение объемов грузоперевозок и стремительное развитие международной логистики. Во-вторых, клиенты становятся все более требовательными, ожидая не только скоростной доставки, но и информацию в режиме реального времени о состоянии своих грузов. В-третьих, необходимость оптимизации маршрутов, управления запасами и сокращения издержек становятся более актуальными, чем когда-либо.

В рамках данного курсового проекта планируется создание базовой версии программного обеспечения, охватывающей основные функциональные элементы логистической компании. Продукт будет разработан с соблюдением современных стандартов разработки программного обеспечения и будет обеспечивать удобный и интуитивно понятный интерфейс для пользователей.

Актуальность данной темы обусловлена ростом объемов грузоперевозок, увеличением ожиданий клиентов и необходимостью оптимизации процессов в логистических компаниях.

Основной целью данного курсового проекта является разработка и реализация программного продукта, который облегчит и улучшит управление логистическими процессами для логистической компании. Для достижения этой цели ставятся следующие задачи:

* проектирование базы данных и архитектуры приложения;
* разработка функционала для отслеживания грузов, сбора статистики, хранения информации о клиентах;
* создание пользовательского интерфейса для клиентов и логистических менеджеров. Данный курсовой проект призван продемонстрировать практическое применение современных технологий в логистической индустрии и подчеркнуть важность развития информационных решений для решения актуальных задач в этой области.

1. Постановка задачи

Целью данной курсовой работы является создание программного обеспечения для логистической компании «Быстрее света», которая позволит вести учёт:

* посылок;
* статусов посылок;
* пунктов выдачи;
* складов;
* машин.

Даст возможность клиентам самостоятельно ознакомиться с услугами и рассчитать стоимость доставки на сайте. А также исключает ошибки, допускаемые при ручной обработке данных.

Для достижения поставленной цели необходимо решить следующие задачи:

* проанализировать предметную область, изучить сферу деятельности логистической компании, для их автоматизации с помощью базы данных;
* написать варианты использования программного обеспечения;
* проанализировать данные, которые будут храниться в базе данных;
* описать логическую структуру базы данных, которая будет описывать как происходит обмен информацией в базе данных;
* выбрать технические и программные средства для реализации проекта;
* описать физическую структуру программы. Описать диаграмму классов разрабатываемого программного обеспечения;
* разработать базу данных для логистической компании «Быстрее света»;
* разработать программное обеспечение для логистической компании «Быстрее света»;
* протестировать разработанное приложение;
* составить руководство пользователя, в котором будут расписаны все основные функциональные возможности разработанного программного обеспечения

В первом разделе «постановка задачи» описывается цель курсового проекта и задачи, которые необходимо выполнить для достижения цели.

Во втором разделе «системный проект» описана предметная область, и логическая структура базы данных.

В третьем разделе «технический проект» происходит выбор программных и технических средств, а также описывается физическая структура базы данных.

В последнем разделе «экспериментальная часть» описан процесс тестирования приложения и его результаты, методики, которые были использованы при тестировании и приведено подробное руководство для пользователя, с описанием всех возможностей разработанного приложения

1. Системный проект
   1. Описание предметной области

Компания «Быстрее света» занимается доставкой посылок. Клиент обладает возможностью отправить посылку при обращении в пункт выдачи. Для отправки посылки необходимо оформить документы у сотрудника пункта. Документы включают в себя акт приема и накладную на товар, для заполнения данных документов клиент должен предоставить паспортные данные, а также указать пункт выдачи, в который необходимо доставить товар.

Сотрудник на основании этих данных должен заполнить акт приема, указав в нем номер пункта, где был принят товар. Сотрудник обязан проверить, что товар соответствует регламентам перевозки, в противном случае сотрудник сообщает клиенту причину, по которой товар не может быть принят.

При оформлении посылки высчитывается стоимость доставки с учетом веса и категории груза. Далее в акт приема вписываются характеристики посылки: вес, габариты и тип упаковки, порядковый номер. Клиент должен заполнить и подписать «Согласие на обработку персональных данных» после оплачивает, ставит подпись в акте приема и получает его копию.

Принятые посылки передаются на перевозку. Перевозка осуществляется при помощи автомобилей компании. После принятия посылок работник пункта выдачи заносит в акт приема со склада полученные посылки, в акт отправки на склад переданные на склад посылки.

При совершении перевозки, на складе в машину загружаются посылки для отправки на определенный склад или в пункт выдачи. После загрузки водитель получает сопроводительные документы, также ему сообщается конечная точка маршрута. Если перевозка осуществлялась в другой регион тогда, после того как водитель доставил посылки, они сортируются для развоза по пунктам выдачи.

При доставке посылки в пункт выдачи клиенту сообщается срок, до которого он должен забрать посылку. В случае если клиент не явился в пункт до озвученного срока, посылка отправляется на утилизацию.

Клиент приходит за заказом. Проверяет его целостность, ставит подпись и забирает товар.

* 1. Диаграммы вариантов использования

Диаграмма вариантов использования - диаграмма, описывающая, какой функционал разрабатываемой программной системы доступен каждой группе пользователей.

Диаграмма вариантов использования показана на рисунке 2.2.1.

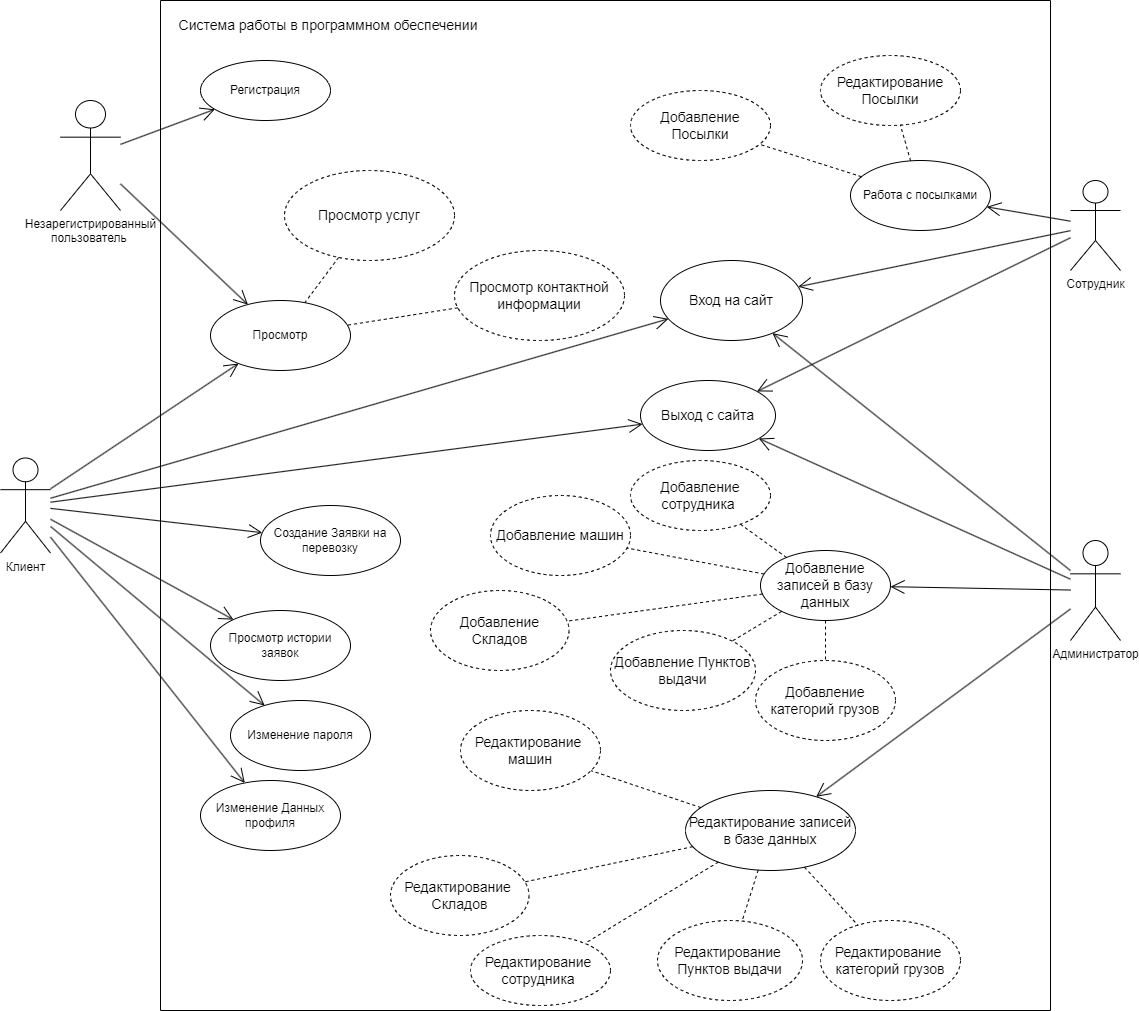


Рисунок 2.2.1 Диаграмма вариантов использования

Возможные действия незарегистрированного пользователя:

* регистрация на сайте;
* просмотр контактной информации;
* просмотр услуг.

Возможные клиента:

* создание заявки на перевозку;
* вход на сайт;
* изменение пароля;
* изменение данных профиля;
* выход с сайта;
* просмотр истории заявок;
* просмотр контактной информации;
* просмотр услуг.

Доступные действия сотрудника:

* вход на сайт;
* добавление посылок;
* редактирование посылок;
* выход с сайта;

Доступные действия администратора:

* вход на сайт;
* добавление пунктов выдачи;
* редактирование пунктов выдачи;
* редактирование категорий грузов;
* добавление категорий грузов;
* добавление машин;
* редактирование машин;
* добавление сотрудника;
* редактирование сотрудника.
  1. Сценарии вариантов использования

Для логистической агентства могут существовать различные сценарии использования сайта:

* + 1. Регистрация на сайте

Краткое описание: клиент хочет зарегистрироваться на сайте.

Основной поток событий:

* пользователь выбирает действие «Зарегистрироваться»;
* система выводит страницу с формой регистрации;
* пользователь вводит данные, такие как имя, фамилия, отчество, номер телефона, почта, серия и номер паспорта, пароль, имя пользователя, дату рождения и выбирает действие «Зарегистрироваться»;
* система проверяет данные из формы и записывает их в базу данных.

Альтернативные потоки:

* при неверном заполнении данных пользователем, выводится сообщение о необходимости исправления тех или иных ошибок.
  + 1. Вход на сайт

Краткое описание: пользователь хочет войти на сайт, для доступа к персонализированным услугам и информации.

Основной поток событий:

* пользователь выбирает действие «Вход»;
* система выводит страницу с формой входа;
* пользователь вводит данные, такие как имя пользователя и пароль и выбирает действие «Вход»;
* система проверяет введенные данные и авторизует, после чего выводит главную страницу;

Альтернативные потоки:

* если пользователь не заполнил какое-либо поле в форме, то в программном обеспечении выводится ошибка с указанием полей, которые нужно заполнить;
* если пользователь забыл пароль, он может воспользоваться функцией сброса пароля, которая отправит ему инструкции на восстановление пароля на указанный адрес электронной почты.

Предусловия: пользователь должен быть зарегистрирован в системе.

* + 1. Добавление информации о сотруднике

Краткое описание: администратор добавляет нового сотрудника.

Основной поток событий:

* администратор выбирает действие «Добавить пользователя»;
* система загружает форму добавления пользователя;
* администратор заполняет данные, такие как имя, фамилия, отчество, номер телефона, почта, серия и номер паспорта, пароль, имя пользователя, дату рождения, роль и выбирает действие «Сохранить»;
* система добавляет информацию о новом сотруднике в базу данных.

Альтернативный поток:

* при неверном заполнении, выводится сообщение о необходимости исправления тех или иных ошибок.

Предусловия:

* для добавления пользователя администратор обязательно должен быть авторизирован.
  + 1. Создание заявки на перевозку

Краткое описание: клиент хочет создать заявку на перевозку.

Основной поток событий:

* клиент выбирает действие «Создать заявку на перевозку»;
* система загружает форму для создания заявки на перевозку;
* клиент вводит данные, такие как адрес отправки, адрес доставки, вес, длина, высота, дата отправки, ширина, комментарии, категория груза, стоимость доставки и выбирает действие «Создать»;
* система обрабатывает запрос и обновляет введённую информацию в базе данных.

Альтернативный поток:

* клиент не верно указывает формат данных для заполнения полей. Программное обеспечение выводит сообщение о неверном заполнении данных, а также предложит повторный ввод и укажет на ошибки.

Предусловия:

* для создания заявки на перевозку, клиент обязательно должен быть авторизирован.
  + 1. Просмотр услуг

Краткое описание: пользователь просматривает услуги компании.

Основной поток событий:

* пользователь выбирает действие «Услуги»;
* система загружает страницу услуг компании;
* пользователь смотрит услуги компании.

Предусловия:

* для просмотра услуг нужно зайти на сайт.
  + 1. Добавление посылки

Краткое описание: сотрудник добавляет информацию о посылке клиента в базу данных.

Основной поток событий:

* сотрудник выбирает действие «Добавление посылки»;
* система загружает форму добавления посылки;
* сотрудник вводит данные, такие как адрес отправки, адрес доставки, вес, длина, высота, дата отправки, ширина, комментарии, категория груза, стоимость доставки, статус и выбирает действие «Сохранить»;
* система делает запрос в базу данных и добавляет информацию о посылке.

Альтернативный поток:

* при неверном заполнении, выводится сообщение о необходимости исправления тех или иных ошибок.

Предусловия:

* для добавления посылки, пользователь должен быть авторизирован.
  + 1. Редактирование посылки

Краткое описание: сотрудник изменяет информацию о посылке клиента.

Основной поток событий:

* сотрудник выбирает конкретную посылку и выбирает действие «Редактирование посылки»;
* система загружает форму редактирования посылки;
* сотрудник изменяет информацию в полях о посылке и выбирает действие «Сохранить»;
* система делает запрос в базу данных и обновляет информацию о посылке.

Альтернативный поток:

- при неверном заполнении, выводится сообщение о необходимости исправления тех или иных ошибок.

Предусловия:

- для редактирования посылки, пользователь должен быть авторизирован.

* + 1. Добавление пункта выдачи

Краткое описание: администратор добавляет информацию о пункте выдачи в базу данных.

Основной поток событий:

* администратор выбирает действие «Добавление пункта выдачи»;
* система загружает форму добавления пункта выдачи;
* администратор вводит данные, такие как регион город, улица, дом, корпус и выбирает действие «Сохранить»;
* система делает запрос в базу данных и добавляет информацию о пункте выдачи.

Альтернативный поток:

- при неверном заполнении, выводится сообщение о необходимости исправления тех или иных ошибок.

Предусловия:

- для добавления пункта выдачи, пользователь должен быть авторизирован.

* + 1. Редактирование пункта выдачи

Краткое описание: администратор изменяет информацию о пункте выдачи.

Основной поток событий:

* администратор выбирает действие «Пункты выдачи»;
* администратор выбирает конкретный пункт выдачи и выбирает действие «Редактирование пункта выдачи»;
* система загружает форму редактирования пункта выдачи;
* администратор изменяет информацию в полях о пункте выдачи и выбирает действие «Сохранить»;
* система делает запрос в базу данных и обновляет информацию о пункте выдачи.

Альтернативный поток:

-при неверном заполнении, выводится сообщение о необходимости исправления тех или иных ошибок.

Предусловия:

-для редактирования пункта выдачи, пользователь должен быть авторизирован.

* + 1. Добавление категорий груза

Краткое описание: администратор добавляет информацию о категории груза в базу данных.

Основной поток событий:

* администратор выбирает действие «Добавление категории груза»;
* система загружает форму добавления категории груза;
* администратор вводит данные, такие как наименование, коэффициент перевозки, комментарии и выбирает действие «Сохранить»;
* система делает запрос в базу данных и добавляет информацию о категории груза.

Альтернативный поток:

-при неверном заполнении, выводится сообщение о необходимости исправления тех или иных ошибок.

Предусловия:

-для добавления категории груза, пользователь должен быть авторизирован.

* + 1. Редактирование пункта выдачи

Краткое описание: администратор изменяет информацию о пункте выдачи.

Основной поток событий:

* администратор выбирает конкретную категорию груза и выбирает действие «Редактирование категории груза»;
* система загружает форму редактирования категории груза;
* администратор изменяет информацию в полях о категории груза и выбирает действие «Сохранить»;
* система делает запрос в базу данных и обновляет информацию о категории груза.

Альтернативный поток:

-при неверном заполнении, выводится сообщение о необходимости исправления тех или иных ошибок.

Предусловия:

-для редактирования категории груза, пользователь должен быть авторизирован.

* + 1. Добавление машины

Краткое описание: администратор добавляет информацию о машине в базу данных.

Основной поток событий:

* администратор выбирает действие «Добавление машины»;
* система загружает форму добавления машины;
* администратор заполняет данные, такие как идентификационный номер, гос. номер, марка, модель, статус, ID\_водителя и выбирает действие «Сохранить»;
* система делает запрос в базу данных и добавляет информацию о машине.

Альтернативный поток:

-при неверном заполнении, выводится сообщение о необходимости исправления тех или иных ошибок.

Предусловия:

-для добавления машины, пользователь должен быть авторизирован.

* + 1. Редактирование машины

Краткое описание: администратор изменяет информацию о машине.

Основной поток событий:

* администратор выбирает конкретную машину и выбирает действие «Редактирование машины»;
* система загружает форму редактирования машины;
* администратор изменяет информацию в полях о машине и выбирает действие «Сохранить»;
* система делает запрос в базу данных и обновляет информацию о машине.

Альтернативный поток:

-при неверном заполнении, выводится сообщение о необходимости исправления тех или иных ошибок.

Предусловия:

-для редактирования машины, пользователь должен быть авторизирован.

* + 1. Добавление склада

Краткое описание: администратор добавляет информацию о складе в базу данных.

Основной поток событий:

* администратор выбирает действие «Добавление склада»;
* система загружает форму добавления склада;
* администратор заполняет данные, такие как регион, город, улица, дом, корпус и выбирает действие «Сохранить»;
* система делает запрос в базу данных и добавляет информацию о складе.

Альтернативный поток:

-при неверном заполнении, выводится сообщение о необходимости исправления тех или иных ошибок.

Предусловия:

-для добавления склада, пользователь должен быть авторизирован.

* + 1. Редактирование склада

Краткое описание: администратор изменяет информацию о складе.

Основной поток событий:

* администратор выбирает конкретный склад и выбирает действие «Редактирование склада»;
* система загружает форму редактирования склада;
* администратор изменяет информацию в полях о складе и выбирает действие «Сохранить»;
* система делает запрос в базу данных и обновляет информацию о складе.

Альтернативный поток:

-при неверном заполнении, выводится сообщение о необходимости исправления тех или иных ошибок.

Предусловия:

-для редактирования складе, пользователь должен быть авторизирован.

* + 1. Редактирование сотрудника

Краткое описание: администратор изменяет информацию о сотруднике.

Основной поток событий:

* администратор выбирает конкретного сотрудника и выбирает действие «Редактирование сотрудника»;
* система загружает форму редактирования сотрудника;
* администратор изменяет информацию в полях о сотруднике и выбирает действие «Сохранить»;
* система делает запрос в базу данных и обновляет информацию о сотруднике.

Альтернативный поток:

-при неверном заполнении, выводится сообщение о необходимости исправления тех или иных ошибок.

Предусловия:

-для редактирования сотрудника, пользователь должен быть авторизирован.

2.3.17 Изменения пароля

Краткое описание: пользователь изменяет пароль.

Основной поток событий:

* пользователь выбирает действие «изменить пароль»;
* система загружает форму изменения пароля;
* пользователь вводит старый и новый пароль и выбирает действие «Сохранить»;
* система делает запрос в базу данных и обновляет информацию о пользователе.

Альтернативный поток:

-при неверном заполнении, выводится сообщение о необходимости исправления тех или иных ошибок.

Предусловия:

-для редактирования пароля, пользователь должен быть авторизирован.

* + 1. Просмотр контактной информации

Краткое описание: пользователь изменяет информацию о сотруднике.

Основной поток событий:

* пользователь выбирает действие «Контакная информация»;
* система загружает страницу контактной информации;
* пользователь просматривает контактную информация.

Альтернативный поток:

-при неверном заполнении, выводится сообщение о необходимости исправления тех или иных ошибок.

Предусловия:

-для редактирования сотрудника, пользователь должен быть авторизирован.

* + 1. выход с сайта – пользователь выходит из своего аккаунта.

Основной поток событий:

* пользователь выбирает действие «Выход»;
* система выводит главную страницу и очищает данные из сессии;

Предусловия:

* для выхода из аккаунта пользователь должен быть авторизирован.
  + 1. Изменение данных профиля

Краткое описание: пользователь редактирует личную информацию.

Основной поток событий:

* при просмотре собственного профиля пользователя, он выбирает действие «Редактировать данные»;
* система выводит страницу с формой редактирования профиля;
* пользователь изменяет какие-либо данные и выбирает действие «Редактировать»;
* система принимает данные из формы и записывает их в базу данных.

Альтернативный поток:

* при неверном заполнении, выводится сообщение о необходимости исправления тех или иных ошибок.

Предусловия:

* для редактирования личной информации пользователь обязательно должен быть авторизирован.
  + 1. Просмотр истории заявок

Краткое описание: клиент просматривает историю заявок.

Основной поток событий:

* клиент выбирает действие «Просмотр истории заявок»;
* система выводит страницу истории заявок;
* Клиент просматривает историю заявок.

Предусловия:

* для просмотра истории заявок пользователь обязательно должен быть авторизирован.

### Описание данных

На основе анализа работы логистической компании, будут сформированы следующие сущности:

* пункт\_Приема\_Выдачи;
* склад;
* посылка;
* категория\_груза;
* машина;
* пользователь.

Описание сущности «Пункт\_Приема\_Выдачи» представлено в

таблице 2.4.1.

Таблица 2.4.1 – Описание данных сущности «Пункт\_Приема\_Выдачи»

|  |  |  |  |
| --- | --- | --- | --- |
| Поле | Тип | Описание | Примечание |
| 1 | 2 | 3 | 4 |
| ID\_пункта | Числовой | Id пункта выдачи | Первичный ключ |
| Регион | Текстовый | Регион, в котором находится пункт выдачи | Максимальный размер 200 |
| Город | Текстовый | Город, в котором находится пункт выдачи | Максимальный размер 200 |
| Улица | Текстовый | Улица, на которой находится пункт выдачи | Максимальный размер 200 |
| Дом | Текстовый | Номер дома | Максимальный размер 10 |
| Корпус | Текстовый | Корпус дома | Максимальный размер 10 |

Описание сущности «Склад» представлено в таблице 2.4.2.

Таблица 2.4.2 – Описание данных сущности «Склад»

|  |  |  |  |
| --- | --- | --- | --- |
| Поле | Тип | Описание | Примечание |
| 1 | 2 | 3 | 4 |
| ID\_склада | Числовой | Id Склада | Первичный ключ |
| Регион | Текстовый | Регион, в котором находится склад | Максимальный размер 200 |
| Город | Текстовый | Город, в котором находится склад | Максимальный размер 200 |

Продолжение таблицы 2.4.2

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 2 | 3 | 4 |
| Улица | Текстовый | Улица, на которой находится склад | Максимальный размер 200 |
| Дом | Текстовый | Номер склада | Максимальный размер 10 |
| Корпус | Текстовый | Корпус склада | Максимальный размер 10 |

Описание сущности «Посылка» представлено в таблице 2.4.3.

Таблица 2.4.3 – Описание данных сущности «Посылка»

|  |  |  |  |
| --- | --- | --- | --- |
| Поле | Тип | Описание | Примечание |
| 1 | 2 | 3 | 4 |
| ID\_посылки | Числовой | Id посылки | Первичный ключ |
| Комментарии | Текстовый | Комментарии к заказу | Необязательный, максимальный размер 500 |
| Вес | Числовой | Вес | Неотрицательный, дробный указывается в килограммах |
| Дата\_принятия | Дата | Дата принятия | Устанавливается дата, когда посылка принята от клиента |
| Дата\_доставки\_в\_ пункт\_выдачи | Дата | Дата доставки в пункт выдачи | Устанавливается дата доставки в пункт выдачи |
| Дата \_выдачи | Дата | Дата выдачи | Не может быть меньше даты принятия |
| Длина | Текстовый | Габариты | Длина указывается в метрах |
| Высота | Текстовый | Габариты | Высота указывается в метрах |
| Ширина | Текстовый | Габариты | Ширина указывается в метрах |
| Стоимость\_доставки | Числовой | Стоимость доставки | Неотрицательная, указывается в рублях, дробный тип. |

Продолжение таблицы 2.4.3

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 2 | 3 | * 4 |
| Статус | Текстовый | Статус | Может иметь следующие значения:   * принят от клиента; * отправлен на склад; * отправлен на склад в город назначения; * принят на складе в городе назначения; * отправлен в пункт выдачи; * принят в пункте выдачи; * утерян; * заявка на перевозку; * утилизирован;   выдан. |

Описание сущности «Категория\_груза» представлено в таблице 2.4.4.

Таблица 2.4.4 – Описание данных сущности «Категория\_груза»

|  |  |  |  |
| --- | --- | --- | --- |
| Поле | Тип | Описание | Примечание |
| 1 | 2 | 3 | 4 |
| ID\_категории | Числовой | Id категории | Первичный ключ |
| Наименование | Текстовый | Наименование | Максимальный размер 200 |
| Коэфициент\_перевозки | Числовой | Коэффициент перевозки | Больше 0, дробный тип с точностью 2. |
| Комментарии | Текстовый | Комментарии | Максимальный размер 500, необязательный |

Описание сущности «Машина» представлено в таблице 2.4.5.

Таблица 2.4.5 – Описание данных сущности «Машина»

|  |  |  |  |
| --- | --- | --- | --- |
| Поле | Тип | Описание | Примечание |
| 1 | 2 | 3 | 4 |
| ID\_машины | Числовой | Id машины | Первичный ключ |
| Идентификационный\_номер | Текстовый | Идентификационный номер машины (VIN) | Уникальное значение, фиксированный размер 17 символов |
| Гос\_номер | Текстовый | Гос. номер машины | Состоит из 8 или 9 символов |
| Марка | Текстовый | Марка машины | Максимальный размер 200 |
| Модель | Текстовый | Модель машины | Максимальный размер 200 |
| Статус | Текстовый | Статус | Может иметь следующие значения:   * на базе; * назначен водитель; * ремонт; * списана. |

Описание сущности «Пользователь» представлено в таблице 2.4.6

Таблица 2.4.6 – Описание данных сущности «Пользователь»

|  |  |  |  |
| --- | --- | --- | --- |
| Поле | Тип | Описание | Примечание |
| 1 | 2 | 3 | 4 |
| ID\_пользователя | Числовой | Id пользователя | Первичный ключ |
| Логин | Текстовый | Логин  Пользователя | Максимальный размер 150 |
| Пароль | Текстовый | Пароль для входа в систему | Максимальный размер 128 |
| Дата\_рождения | Дата | Дата рождения пользователя |  |

Продолжение таблицы 2.4.6

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 2 | 3 | 4 |
| E-mail | Текстовый | E-mail | Максимальный размер 254, уникальное значение |
| Имя | Текстовый | Имя | Максимальный размер 50 |
| Фамилия | Текстовый | Фамилия | Максимальный размер 50 |
| Отчество | Текстовый | Отчество | Необязательный, Максимальный размер 50 |
| Серия\_и\_номер\_ паспорта | Текстовый | Серия и номер паспорта | Уникальное значение, фиксированный размер 10 |
| Телефон | Текстовый | Номер телефона | Максимальный размер 11 |
| Роль | Текстовый | Роль Пользователя | Может иметь следующие значения:   * клиент; * сотрудник; * водитель; * администратор. |

## 2.5 Логическая структура базы данных

Логическая структура базы данных — это схема, которая демонстрирует логические ограничения, которые распространяются на хранимые данные. [В ней отражаются, представления](https://www.lucidchart.com/pages/ru/%D1%81%D1%85%D0%B5%D0%BC%D1%8B%D0%B1%D0%B0%D0%B7-%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) и условия целостности .

[Под логической структурой понимается совокупность сущностей, содержащихся в них полей и связей между этими сущностями, с которой взаимодействует пользовательская программа, обрабатывающая базу данных](https://it.wikireading.ru/41761).

Логическая структура базы данных логистической компании изображена на рисунке 2.5.1.

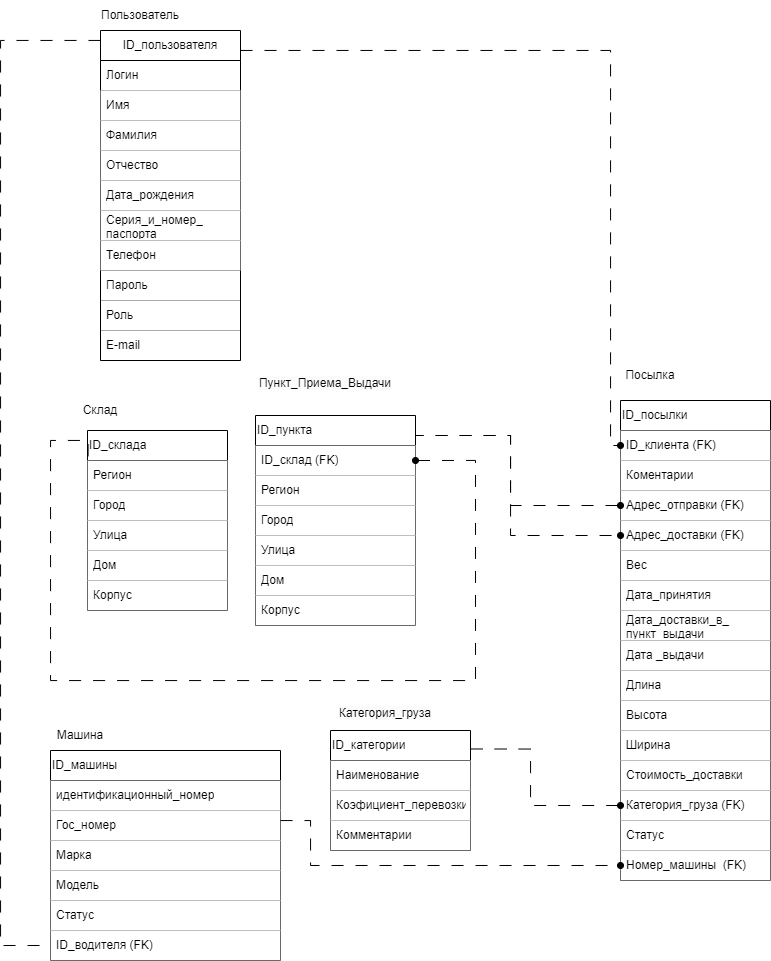


Рисунок 2.3.1 – логическая структура базы данных

Рисунок 2.3.1 – логическая структура базы данных

Сущность «Пункт\_Приема\_Выдачи» содержит информацию о пунктах, где клиенты могут получить или отдать посылку на перевозку.

Сущность «Склад» содержит информацию о складах, на которых хранятся посылки для дальнейшей перевозки.

Сущность «Посылка» содержит данные посылок, которые перевозит компания.

Сущность «Категория\_груза» содержит категории грузов, которые может перевозить компания

Сущность «Машина» хранит информацию о машинах логистической компании.

Сущность «Пользователь» содержит информацию о пользователях сайта, клиентах и сотрудниках.

Связь между сущностями «Пользователь» и «Посылка» выбрана один ко многим, так как один клиент может отправлять несколько посылок.

Связь между сущностями «Машина» и «Посылка» выбрана один ко многим, так как на одной машине может перевозится множество посылок, связь, не идентифицирующая так как у посылка может находиться не машине, а на складе.

Связь между сущностями «Пункт\_Приема\_Выдачи» и «Посылка» выбрана один ко многим, из-за того, что на в одном пункте храниться множество посылок.

Связь между сущностями «Машина» и «Пользователь» выбрана один к одному, так как за одной машиной может быть закреплен только один сотрудник, связь, не идентифицирующая, так как за машиной не обязательно закреплен сотрудник.

Связь между сущностями «Склад» и «Пункт\_Приема\_Выдачи» выбрана один ко многим, так как на один склад привозятся посылки с нескольких пунктов выдачи.

Связь между сущностями «Пользователь» и «Посылка» выбрана один ко многим, так как сотрудник может принять множество посылок.

Связь между сущностями «Категория\_груза» и «Посылка» выбрана один ко многим, так как разные посылки могут иметь одну и туже категорию.

1. Системный проект

3.1 Выбор состава технических и программных средств

Для разработки программного обеспечения и базы данных для логистической компании были выбраны: среда разработки Pycharm, в которой был использован язык программирования Python, фреймворк Django, система управления реляционными базами данных SQLite, также HTML и CSS для размещения на веб-странице различных элементов.

Django – это фреймворк для веб-разработки, который предоставляет инструменты и структуру для создания веб-приложений на языке программирования Python. Django был разработан с целью упростить и ускорить процесс создания веб-приложений, предоставляя готовые решения для многих распространенных задач, связанных с веб-разработкой.

Преимущества Django:

* административная панель: Django поставляется с встроенной административной панелью, которую можно легко настроить для управления данными и администрирования приложения без необходимости создавать пользовательский интерфейс;
* шаблоны и статика: Django позволяет организовать структуру шаблонов и обработку статических файлов, что упрощает разработку пользовательского интерфейса.;
* безопасность: Django включает в себя множество встроенных механизмов безопасности, таких как защита от атак инъекции SQL и других типов атак.

PyCharm – это интегрированная среда разработки (IDE) для языка программирования Python, разработанная компанией JetBrains. Она предоставляет разработчикам множество инструментов и функций, которые упрощают и улучшают процесс создания и поддержки приложений на Python. PyCharm имеет следующие преимущества:

* интеграция с Git: PyCharm интегрирован с системой контроля версий Git, что позволяет управлять исходным кодом и коммитить изменения непосредственно из IDE;
* мощный редактор кода: PyCharm предоставляет богатый редактор кода с автодополнением, подсветкой синтаксиса, проверкой ошибок и другими инструментами, которые упрощают написание и редактирование Python-кода.

Python – это высокоуровневый интерпретируемый язык программирования, который был создан в конце 1980-х годов и развивается до сих пор. Python отличается чистым и читаемым синтаксисом, что делает его одним из наиболее популярных языков программирования в мире. Он имеет следующие преимущества:

* простота и читаемость: Синтаксис Python разработан так, чтобы быть легко читаемым и понятным. Это делает язык доступным для начинающих разработчиков и упрощает поддержку и сопровождение кода.
* кроссплатформенность: Python поддерживается на различных операционных системах, включая Windows, macOS и различные дистрибутивы Linux, что делает его переносимым между платформами;
* динамическая типизация: Python предоставляет динамическую типизацию, что упрощает разработку и уменьшает количество кода, необходимого для решения задач.

SQLite – это компактная и встроенная система управления реляционными базами данных, которая позволяет хранить, управлять и извлекать данные в локальных или встроенных в приложение базах данных. SQLite имеет следующие преимущества: простота в использовании и компактность.

HTML – это язык разметки, используемый для создания структуры и содержания веб-страниц. С помощью HTML разработчики определяют, какие элементы должны присутствовать на веб-странице и как они должны взаимодействовать друг с другом.

CSS – это язык стилей, который используется для определения внешнего вида и макета веб-страниц. CSS позволяет разработчикам задавать цвета, шрифты, размеры, отступы, расположение элементов и другие стилевые атрибуты, которые делают веб-страницу более привлекательной и удобной для пользователей.

3.2 Физическая структура базы данных

Физическая структура базы данных включает в себя описание файлов и групп файлов данных, журнала транзакций, первоначальный размер файлов, шаг прироста базы данных, ее максимальный размер, параметры конфигурации и другие физические характеристики.

Физическая структура базы данных логистической компании изображена на рисунке 3.2.1.

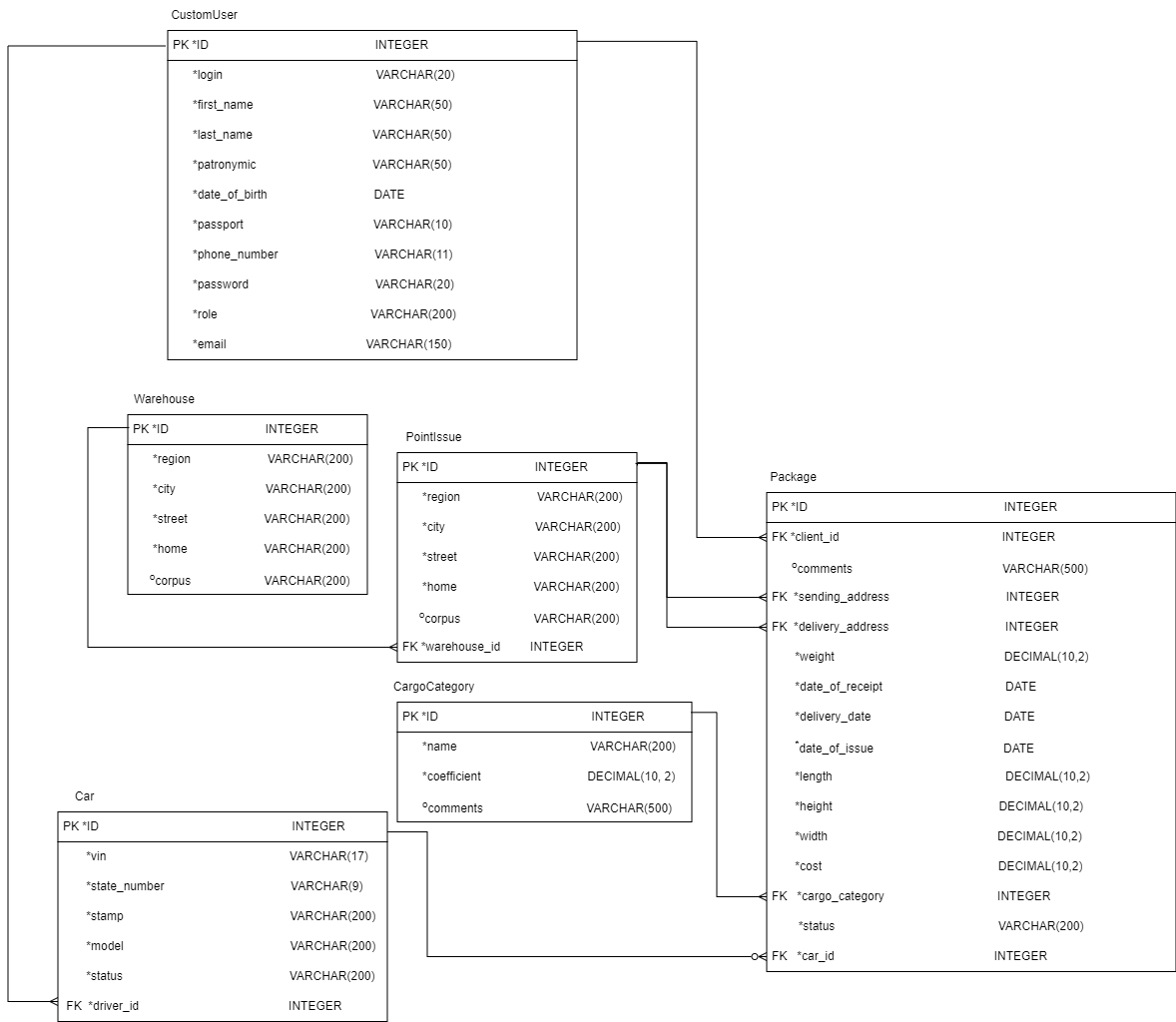


Рисунок 3.2.1 Физическая структура базы данных

Ключевое слово «UNIQUE» используется в таблице Car для определения уникальности значения в поле vin. Оно гарантирует, что каждое значение в указанном столбце будет уникальным, то есть не будет дублироваться в пределах таблицы.

Для всех полей, кроме corpus, driver\_id, delivery\_date, date\_of\_issue car\_id, comments в таблицах package, pointIssue, car, Warehouse, cargoCategory, указывается ключевое слово «NOT NULL» для определения ограничения на поля, которое требует, чтобы в этих полях всегда присутствовало непустое значение. Определение поля с ограничением "NOT NULL" гарантирует, что каждая запись в таблице будет содержать непустое значение в указанном поле. Это используется для обеспечения целостности данных.

3.3 Физическая структура программы

Запуск программного обеспечения начинается с файла «manage.py», который находится в корне проекта.

В папке «kursovaya» расположены файлы приложения, такие как:

* \_\_init\_\_.py, пустой файл для того, чтобы Django и Python распознавали папку как Python модуль;
* asgi.py, интерфейс шлюза асинхронного сервера;
* settings.py, содержит в себе все настройки проекта. Здесь регистрируется приложения, задаётся размещение статичных файлов, настройки базы данных и так далее;
* urls.py, задаёт ассоциации url адресов с представлениями;
* wsgi.py, используется для налаживания связи между Django приложением и веб-сервером.

В папке «pages» расположены файлы приложения, такие как:

* \_\_init\_\_.py, пустой файл для того, чтобы Django и Python распознавали папку как Python модуль;
* admin.py, используется для регистрации моделей в вашем приложении в администрации Django;
* apps.py, это общий файл конфигурации для всех приложений Django. С помощью этого файла можно настроить атрибуты для своего приложения;
* forms.py, используется для определения форм Django, которые используются для ввода данных пользователем. ;
* models.py, в этом файле определяются модели данных Django для приложения;
* urls.py, задаёт ассоциации url адресов с представлениями;
* views.py, содержит представления, которые обрабатывают логику запросов и ответов в приложении.

Полная структура вышеперечисленных каталогов изображена на рисунке 3.3.1

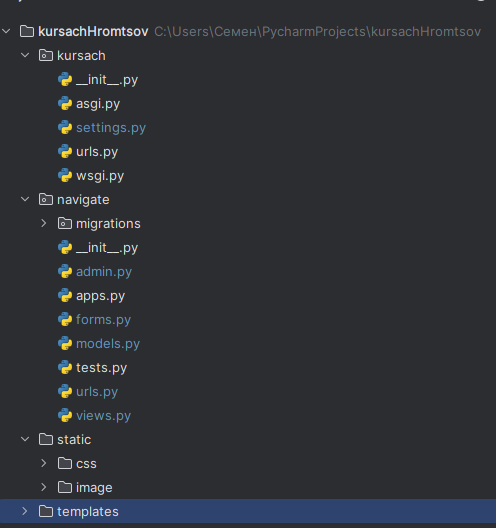


Рисунок 3.3.1 – Иерархия папок в проекте

Для реализации работы с базой данных были созданы классы-модели:

* CustomUser;
* CargoCategory;
* Warehouse;
* Car;
* PointIssue;
* Package;

Диаграмма классов изображена на рисунке 3.3.2.

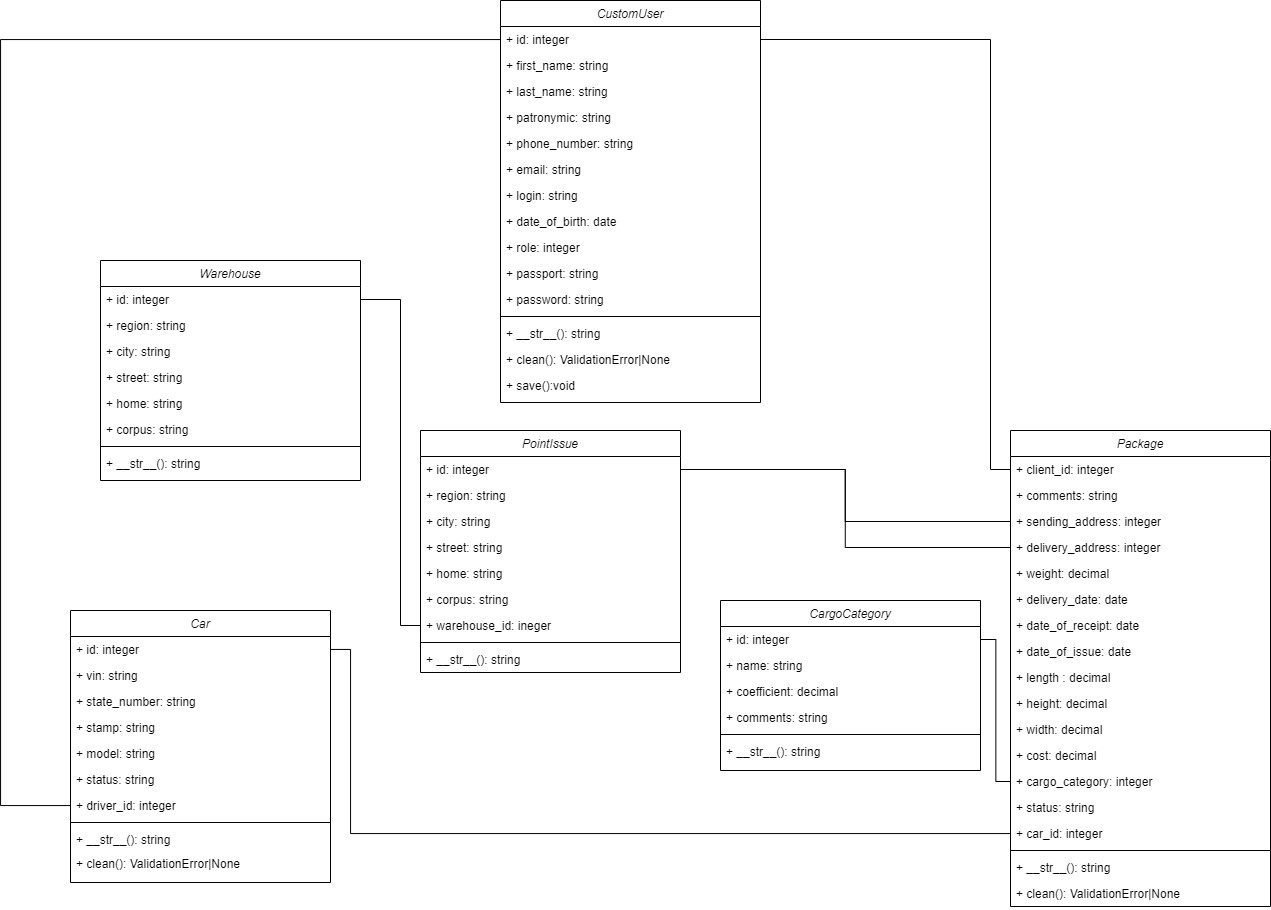


Рисунок 3.3.2 – Диаграмма классов

# Экспериментальная часть

## Тестирование приложения

Для тестирования приложения было выбрано ручное тестирование.

Ручное тестирование приложения — это процесс проверки программного обеспечения человеком, тестировщик проверяет приложение, используя его так, как это делал бы обычный пользователь.

Преимущества ручного тестирования:

* тестировщик выполняет роль обычного пользователя и очень подробно изучает весь функционал и особенности сайта;
* тестирование в реальном времени. Незначительные изменения могут быть исследованы сразу, без написания кода и его исполнения.
* тестирования пользовательского интерфейса;

После выбора вида тестирования были проверены следующие сценарии:

4.1.1 Сценарий удачной регистрации

Заполнить все поля корректными данными, нажать на кнопку «Зарегистрироваться» и успешно зарегистрировать учетную запись, после чего должна произойти переадресация на страницу входа.

4.1.2 Сценарий неудачной регистрации

Заполнить не все поля, некорректными данными или адрес электронной почты или логин, который используется другим пользователем. Получать следующие возможные сообщения об ошибках (рисунок 4.1.2.1):

* «Пользователь с таким Адрес электронной почты уже существует»;
* «Ваш пароль должен содержать как минимум 8 символов»;
* «Обязательное поле»;
* «Пользователь с таким Имя пользователя уже существует»;
* «Номер телефона должен содержать только 11 цифр»;
* «Паспорт должен содержать 10 цифр»;
* «Имя/Фамилия/Отчество должны содержать кириллицу».

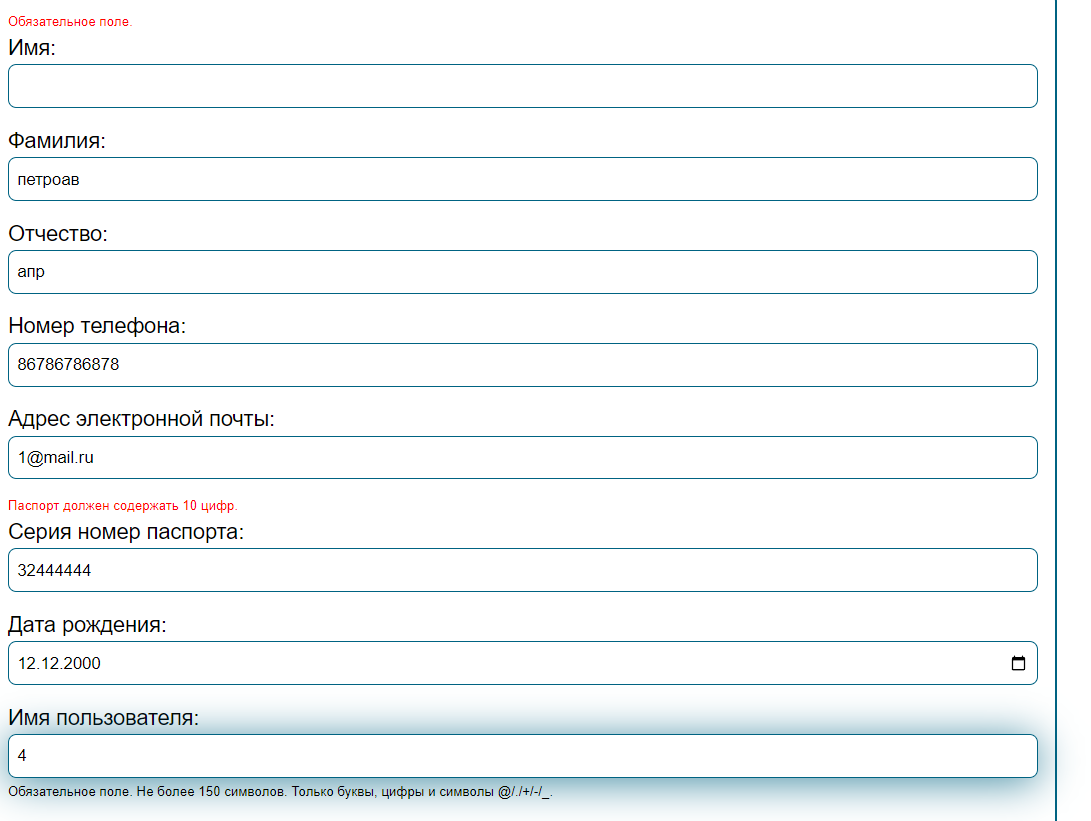


Рисунок 4.1.2.1 – Ошибки при неудачной регистрации

4.1.3 Сценарий удачной авторизации

Заполнить данные существующего пользователя и нажать на кнопку «Войти», после чего осуществляется авторизация и должна произойти переадресация на главную страницу. В шапке отображаются кнопки личного кабинета и «Выйти».

4.1.4 Сценарий неудачной авторизации

Заполнить данные несуществующего пользователя и получать сообщение об ошибке: «Пожалуйста, введите правильные Логин и пароль. Оба поля могут быть чувствительны к регистру.» (рисунок 4.1.4.1).

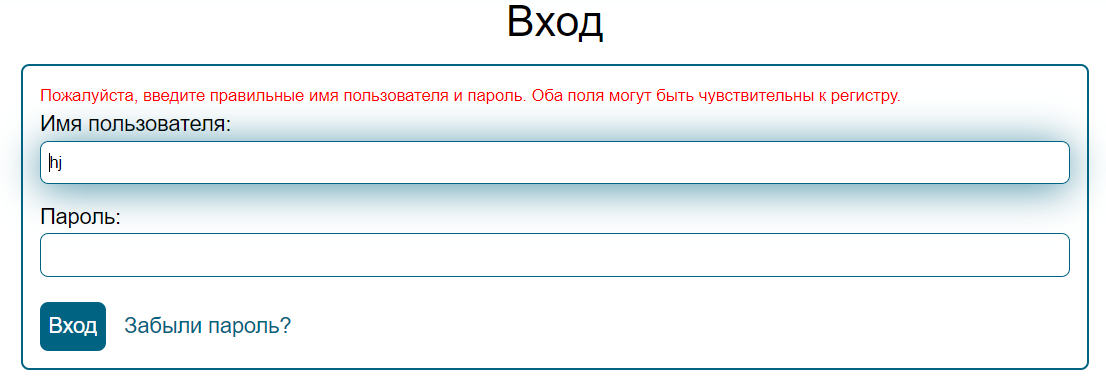


Рисунок 4.1.4.1 – Ошибки при неудачной попытке входа

4.1.5 Сценарий выхода из учетной записи

Нажать на кнопку «Выйти» в навигационном меню, после чего осуществляется выход из аккаунта и должна произойти переадресация на главную страницу. В шапке отображаются кнопки «Войти» и «Регистрация».

4.1.6 Сценарий просмотра услуг

Нажать на кнопку «Услуги» в навигационном меню, после чего должна произойти переадресация на страницу услуг, где можно отследить посылку и сделать заявку на отправку посылки.

4.1.7 Сценарий просмотра контактной информации

Нажать на кнопку «Контакты» в навигационном меню, после чего должна произойти переадресация на страницу контактов.

4.1.8 Сценарий удачного создания заявки на перевозку

Нажать на кнопку «Услуги» в навигационном меню, после чего должна произойти переадресация на страницу услуг, где нажать на кнопку «Рассчитать стоимость и создать заявку на перевозку», заполнить корректными данными, нажать на кнопку «Создать», должна произойти переадресация на страницу «Мои посылки».

4.1.9 Сценарий неудачного создания заявки на перевозку

Нажать на кнопку «Услуги» в навигационном меню, после чего должна произойти переадресация на страницу услуг, где нажать на кнопку «Рассчитать стоимость и создать заявку на перевозку», заполнить некорректными данными, нажать на кнопку «Создать», получить ошибки: «Введите все числовые данные», «Обязательное поле», «Адрес отправки не должен совпадать с адресом получения».

4.1.10 Сценарий изменения пароля

Нажать на кнопку «Забыли пароль?», после чего должна произойти переадресация на страницу смены пароля, где вводить почту и изменять пароль, после должна произойти переадресация на страницу входа.

4.1.11 Сценарий успешного изменения данных профиля

Нажать на кнопку профиля, после чего должна произойти переадресация на страницу профиля, нажать кнопку «Редактировать», где вводить корректные данные и нажимать «Сохранить», после должна произойти переадресация на страницу профиля.

4.1.12 Сценарий неудачного изменения данных профиля

Нажать на кнопку профиля, после чего должна произойти переадресация на страницу профиля, нажать кнопку «Редактировать», где вводить некорректные данные и нажимать «Сохранить», после получать ошибки.

4.1.13 Сценарий просмотра истории заявок

Нажать на кнопку «Мои посылки», после чего должна произойти переадресация на страницу истории заявок.

4.1.14 Сценарий успешного добавления или редактирования посылок

Переходить на страницу с формой добавления или редактирования посылок, после чего должна произойти переадресация на страницу с формой создания посылки, где вводить корректные данные и нажимать «Сохранить», после должна произойти переадресация на страницу со списком всех посылок.

4.1.15 Сценарий неудачного добавления или редактирования посылок

Переходить на страницу с формой добавления или редактирования посылок, после чего должна произойти переадресация на страницу с формой создания посылки, где вводить некорректные данные и нажимать «Сохранить», после получать ошибки.

4.1.16 Сценарий успешного добавления или редактирования пункта выдачи

Переходить на страницу с формой добавления или редактирования пункта выдачи, после чего должна произойти переадресация на страницу с формой создания посылки, где вводить корректные данные и нажимать «Сохранить», после должна произойти переадресация на страницу со списком всех пунктов выдачи.

4.1.17 Сценарий неудачного добавления или редактирования пункта выдачи

Переходить на страницу с формой добавления или редактирования пункта выдачи, после чего должна произойти переадресация на страницу с формой создания посылки, где вводить некорректные данные и нажимать «Сохранить», после получать ошибки.

4.1.18 Сценарий успешного добавления или редактирования машины

Переходить на страницу с формой добавления или редактирования машины, после чего должна произойти переадресация на страницу с формой создания посылки, где вводить корректные данные и нажимать «Сохранить», после должна произойти переадресация на страницу со списком всех машин.

4.1.19 Сценарий неудачного добавления или редактирования машины

Переходить на страницу с формой добавления или редактирования машины, после чего должна произойти переадресация на страницу с формой создания посылки, где вводить некорректные данные и нажимать «Сохранить», после получать ошибки.

4.1.20 Сценарий успешного добавления или редактирования пользователя

Переходить на страницу с формой добавления или редактирования пользователя, после чего должна произойти переадресация на страницу с формой создания посылки, где вводить корректные данные и нажимать «Сохранить», после должна произойти переадресация на страницу со списком всех пользователей.

4.1.21 Сценарий неудачного добавления или редактирования пользователя

Переходить на страницу с формой добавления или редактирования пользователя, после чего должна произойти переадресация на страницу с формой создания посылки, где вводить некорректные данные и нажимать «Сохранить», после получать ошибки.

## Руководство пользователя

При запуске приложения пользователь видит главную страницу, где будет информация о преимуществах компании. Данная страница представлена на рисунках 4.2.1, 4.2.2.

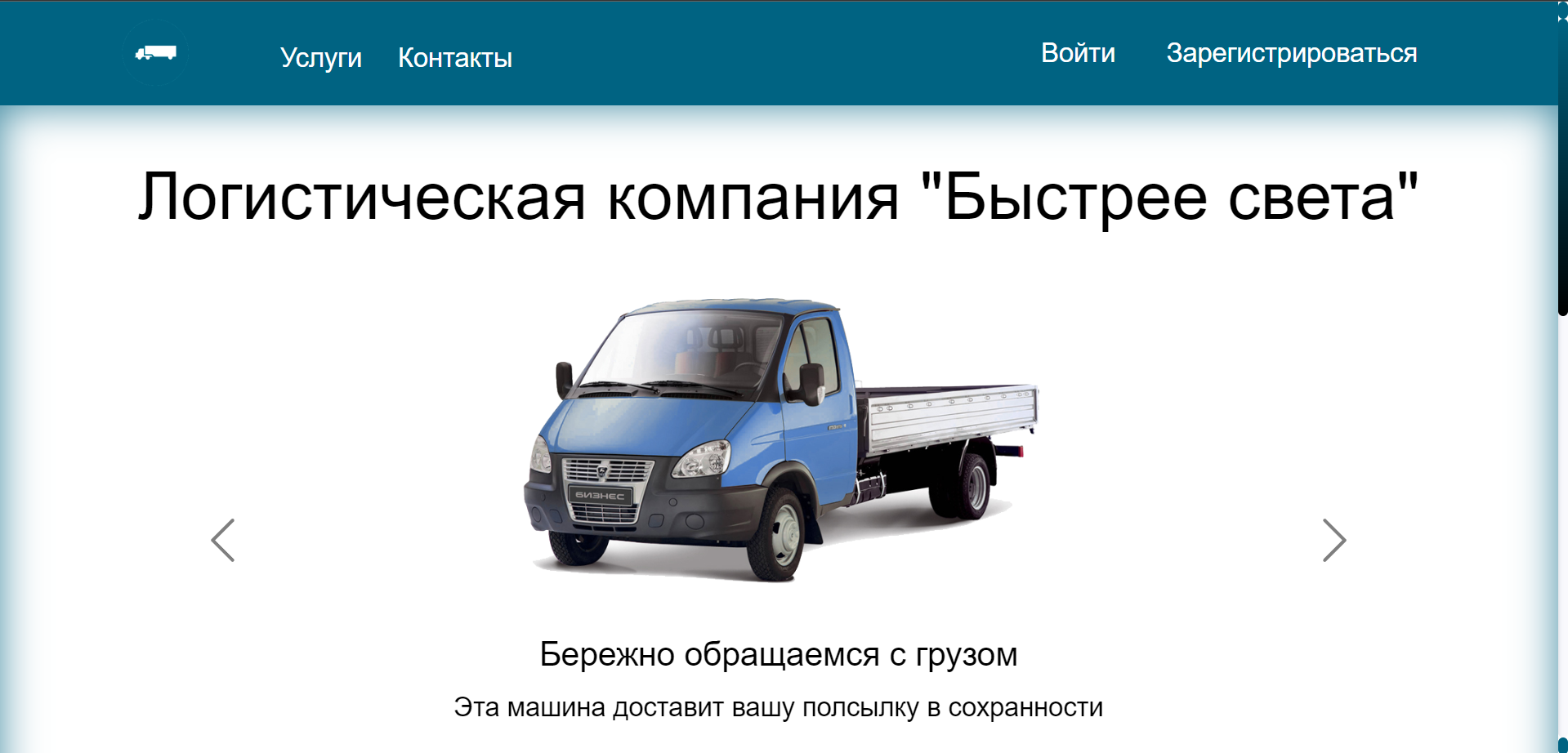


Рисунок 4.2.1 - Главная страница с навигационной панелью

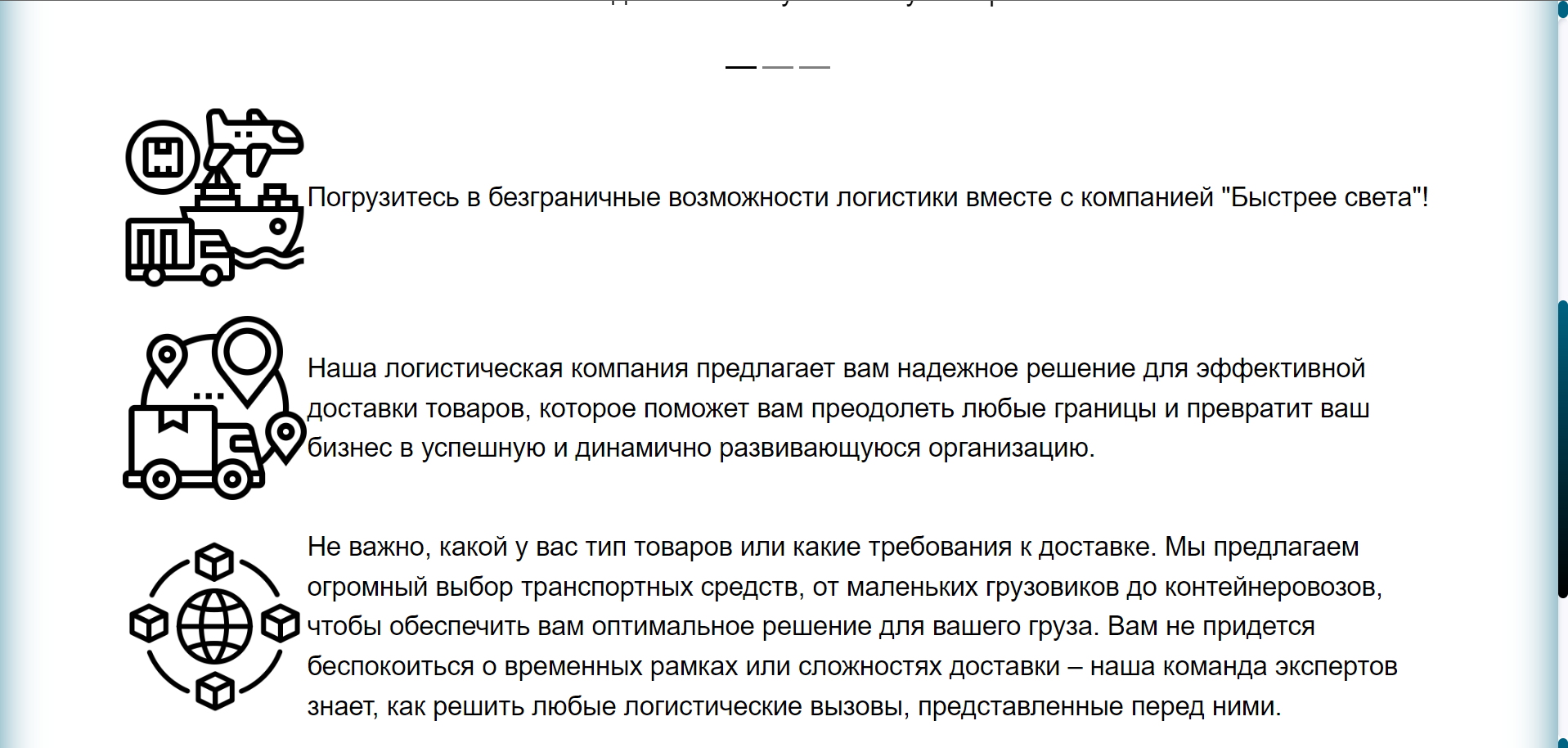


Рисунок 4.2.2 - Главная страница «Наши преимущества»

Для регистрации пользователь нажимает на кнопку «Зарегистрироваться» в навигационном меню приложения, показано на рисунке 4.2.1, система выводит страницу с формой регистрации (рисунок 4.2.3). После заполнения формы пользователь нажимает на кнопку «Зарегистрироваться» приложение принимает данные, регистрирует и выводит страницу с формой входа (рисунок 4.2.4).

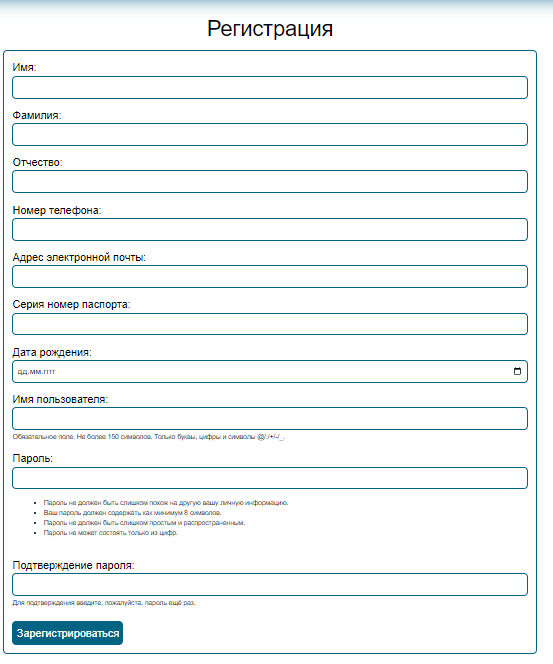


Рисунок 4.2.3 - Страница с формой регистрации

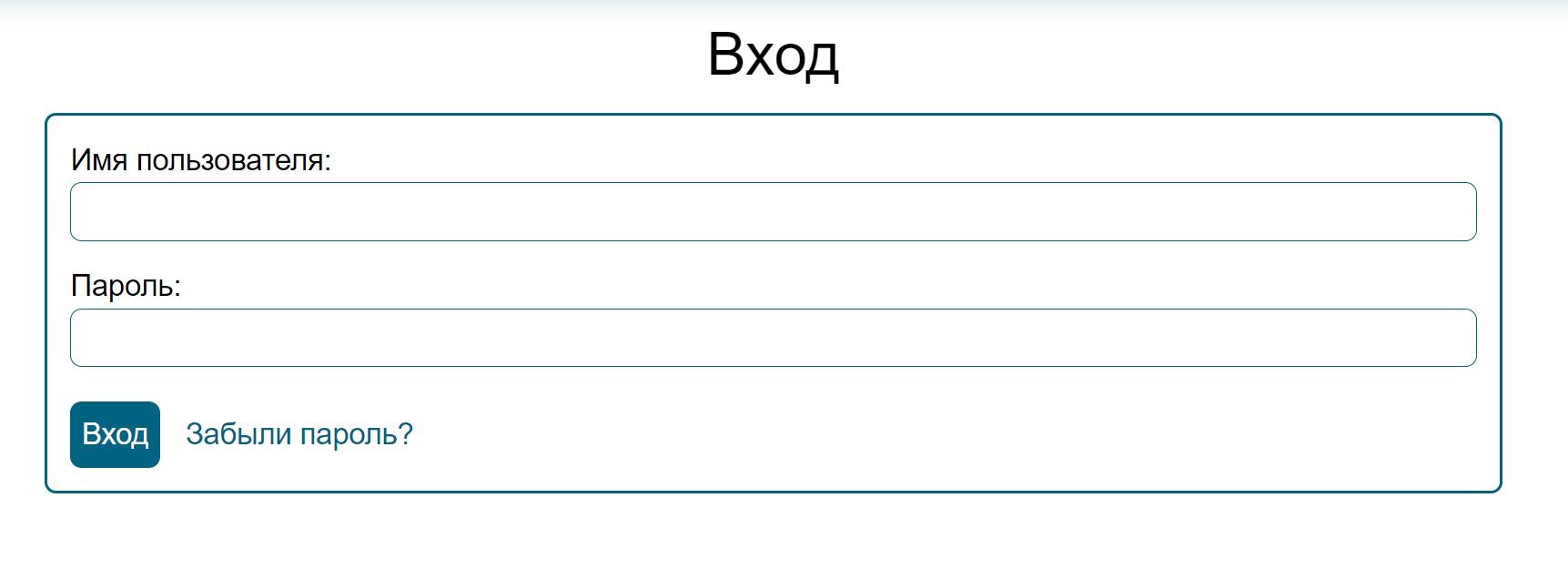


Рисунок 4.2.4 - Страница с формой входа

Помимо регистрации в навигационном меню, показанным на рисунке 4.2.1, есть кнопка «Вход», предназначенная для случая, если у пользователя уже есть учетная запись в системе. При нажатии на неё пользователя выводит страницу с формой входа (рисунок 4.2.4). После нажатия на кнопку «Войти» приложение авторизует пользователя и перенаправит на главную страницу (рисунок 4.2.5).

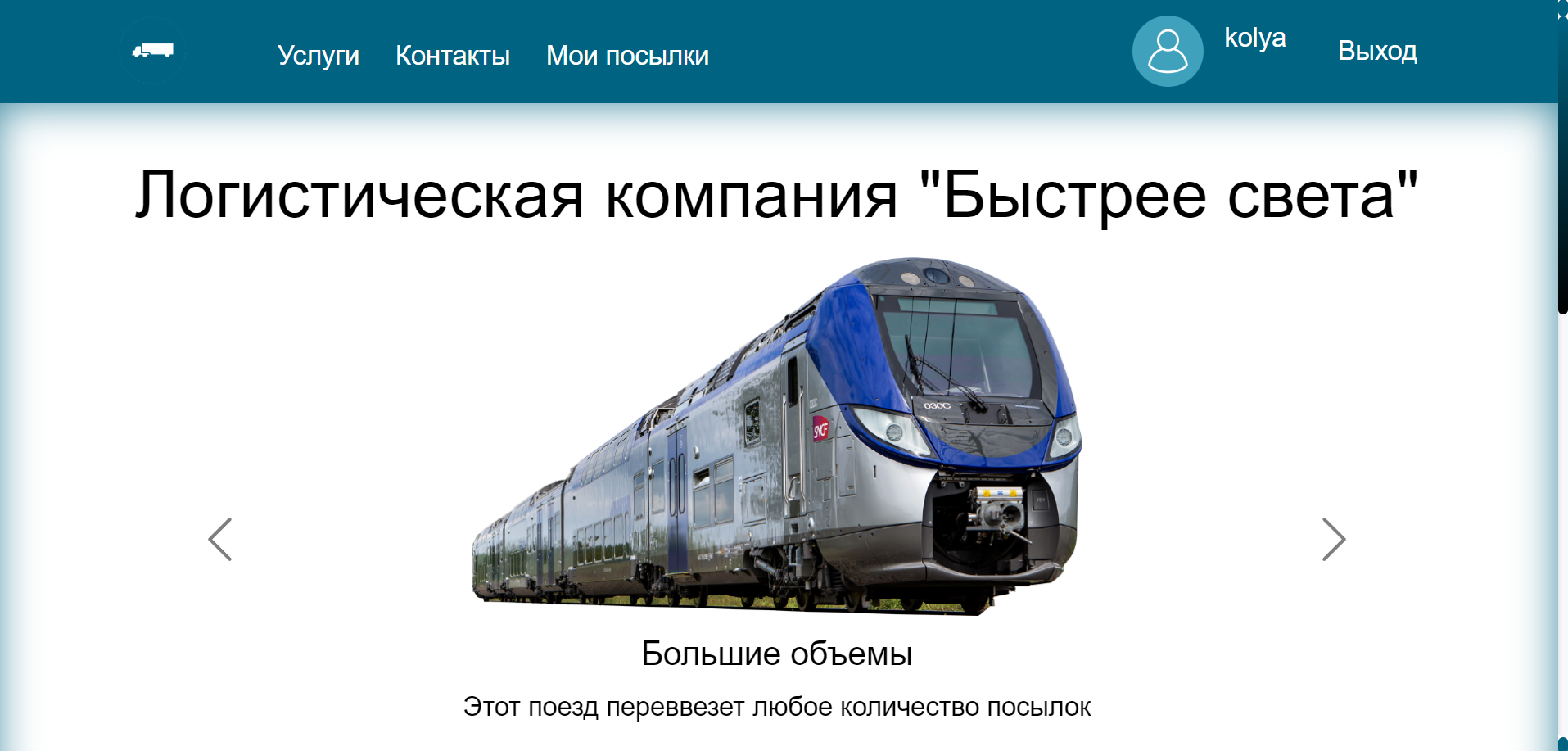


Рисунок 4.2.4 – Главная страница после авторизации пользователя

В навигационном меню, показанным на рисунке 4.2.4, есть кнопка «Мои посылки», предназначенная для перехода на страницу со списком посылок, где пользователь может перейти на страницу с подробным описанием посылки, нажав на кнопку «подробнее» в нужной строке (рисунок 4.2.5).



Рисунок 4.2.5 – Страница со списком посылок

При нажатии на кнопку «Подробнее» на странице посылок, система выводит страницу с подробной информацией посылки (рисунок 4.2.6).

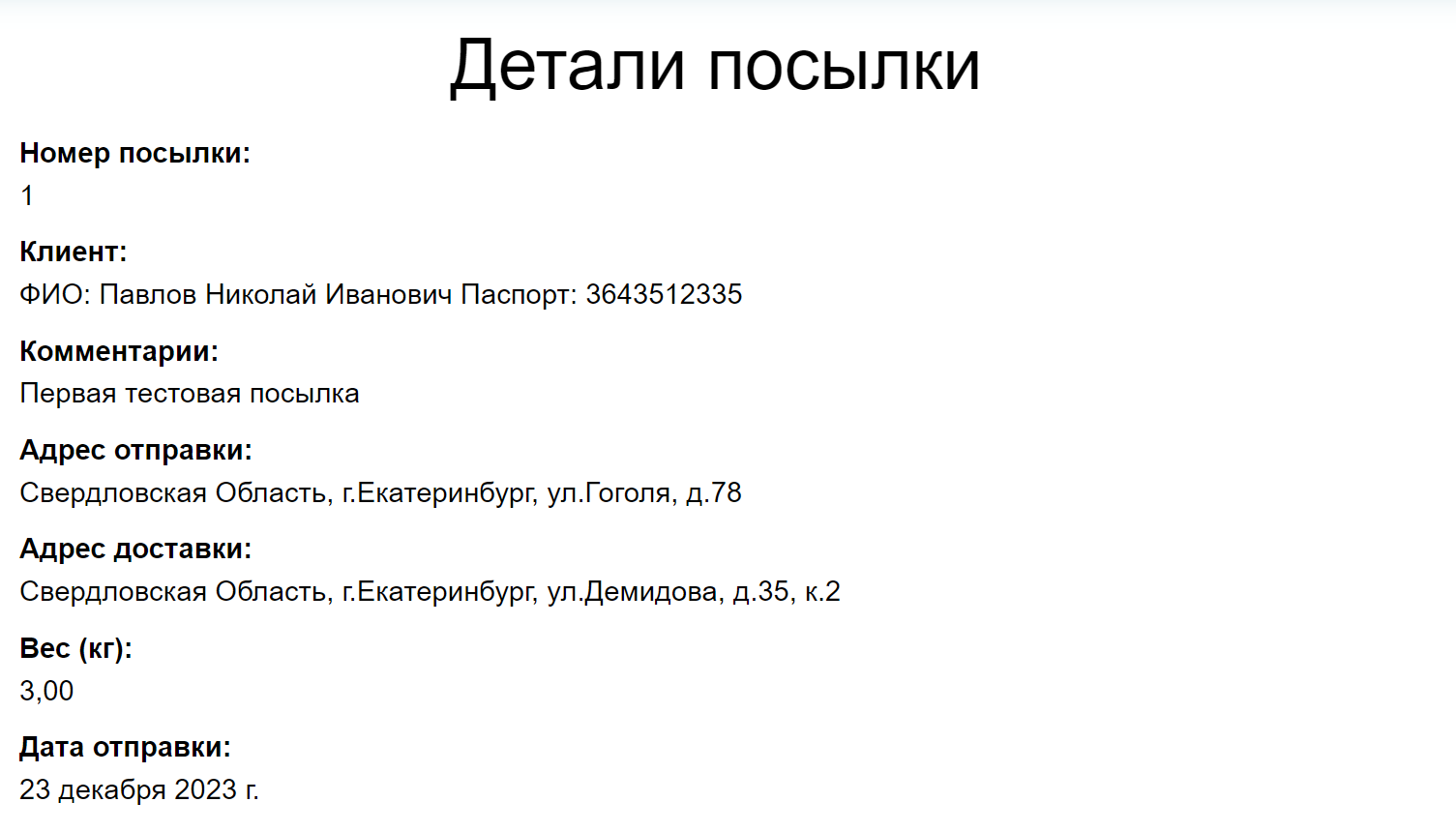


Рисунок 4.2.6 – Страница с подробным описанием посылки

При нажатии на кнопку «Услуги» в навигационном меню, система выводит страницу с услугами (рисунок 4.2.7).



Рисунок 4.2.7 – Страница с услугами

При нажатии на кнопку «Контакты» в навигационном меню, система выводит страницу с контактами (рисунок 4.2.8).



Рисунок 4.2.8 – Страница с контактами

При вводе номера посылки и нажатии кнопки «Отследить» в услугах, мы получаем информацию о посылке (рисунок 4.2.9).



Рисунок 4.2.9 – Страница с информацией о посылке

При нажатии кнопки «Рассчитать стоимость и создать заявку на перевозку» в услугах, система выводит форму с добавлением заявки на перевозку (рисунок 4.2.10).

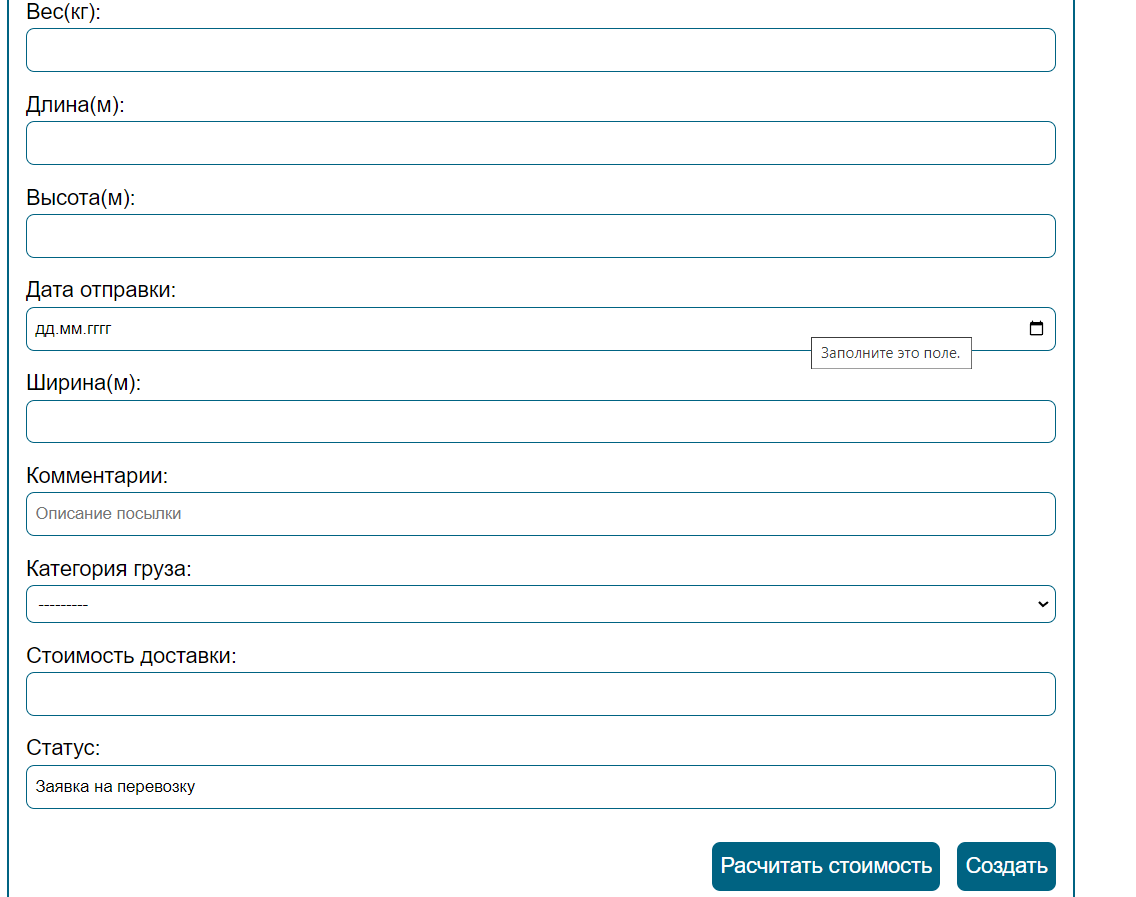


Рисунок 4.2.10 – Страница с формой создания заявки на перевозку

При нажатии кнопки «Рассчитать стоимость», происходит расчет стоимости перевозки посылки

При нажатии кнопки «Создать», при верном вводе данных, система выводит список посылок пользователя (рисунок 4.2.5)

При нажатии на личный кабинет, система выводит страницу с личными данными пользователя (рисунок 4.2.11).

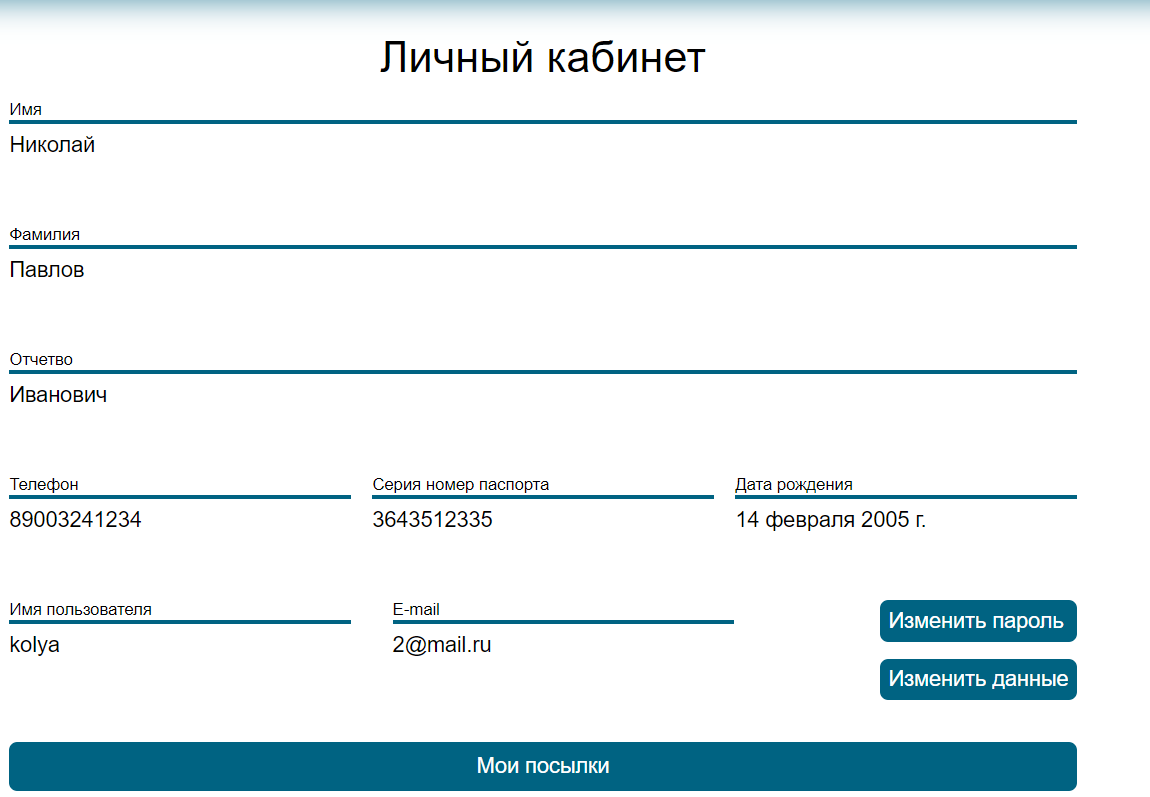


Рисунок 4.2.11 – Страница с личными данными пользователя

При нажатии на кнопку «Изменить данные» в личном кабинете, система выводит форму с данными пользователя (рисунок 4.2.12).



Рисунок 4.2.12 – Страница с формой изменения данных пользователя

Если пользователь является сотрудником, и он авторизован, то навигационное меню выглядит так (рисунок 4.2.13)

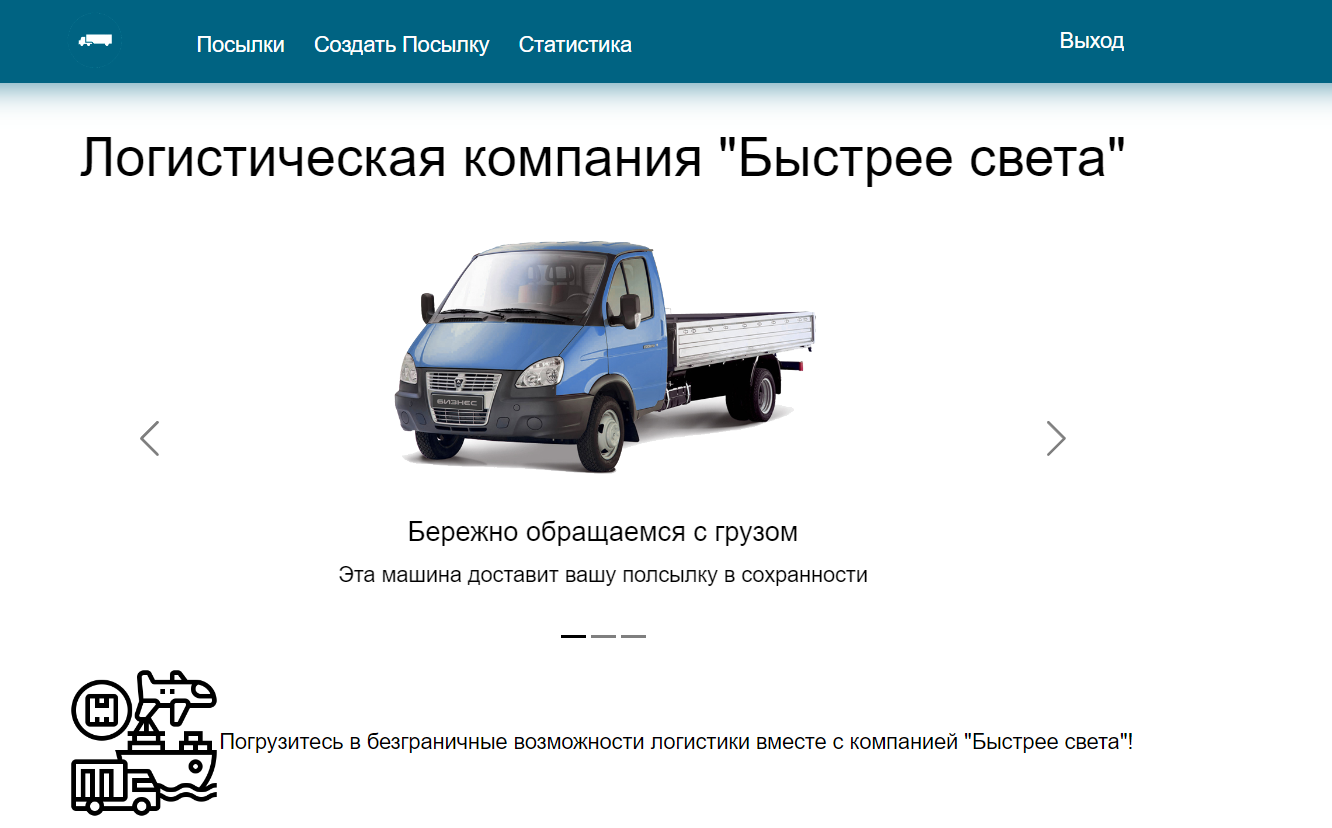


Рисунок 4.2.13 – Навигационное меню сотрудника

При нажатии на кнопку «Создать посылку» в навигационном меню, система выводит форму для создания посылки (рисунок 4.2.14).



Рисунок 4.2.14 – Страница с формой создания посылки

При нажатии на кнопку «Статистика» в навигационном меню, система выводит страницу со статистикой за определенный период (рисунок 4.2.15).

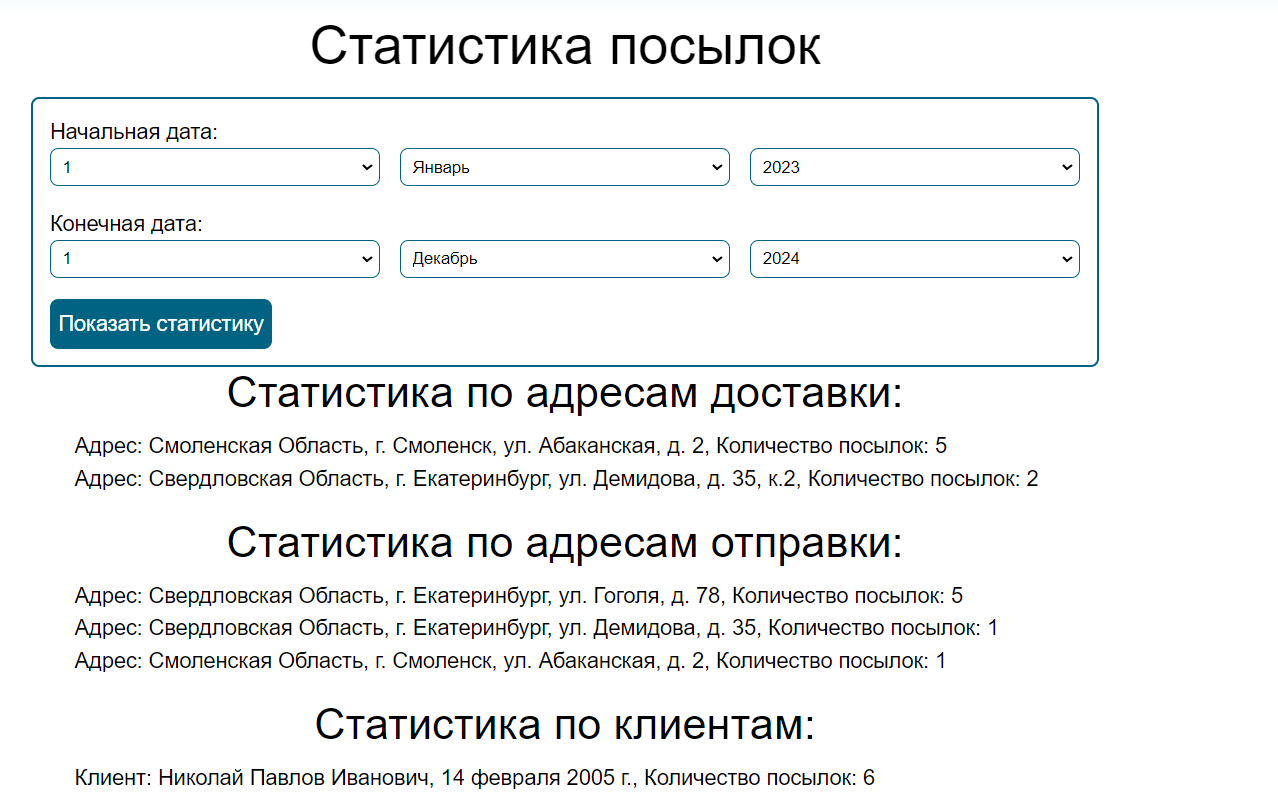


Рисунок 4.2.15 – Страница со статистикой

Если пользователь является администратором, и он авторизован, то админ-панель выглядит так (рисунок 4.2.16)

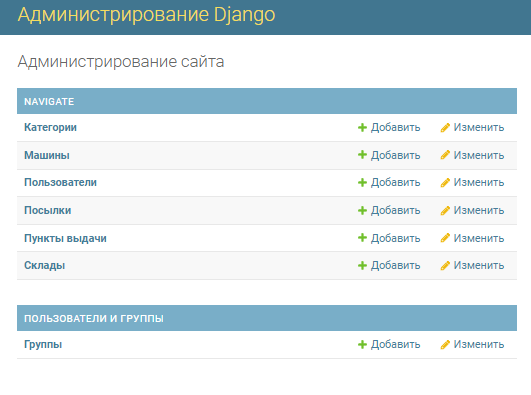


Рисунок 4.2.16 – админ-панель для администратора

При нажатии на кнопку «Добавить» в админ-панели, система выводит форму для создания пункта выдачи (рисунок 4.2.17).

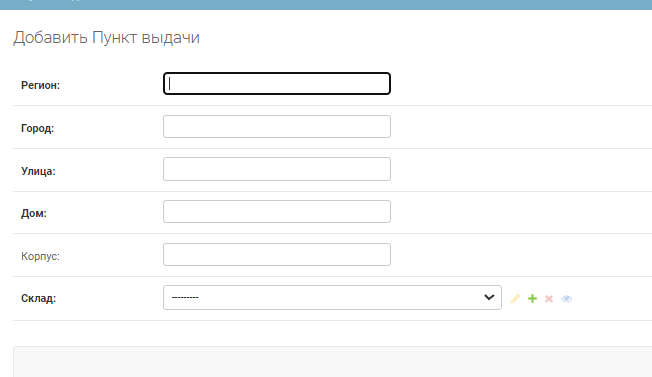


Рисунок 4.2.17 – Страница с формой создания пункта выдачи

При нажатии на кнопку «Добавить» в админ-панели, система выводит форму для создания машины (рисунок 4.2.18).

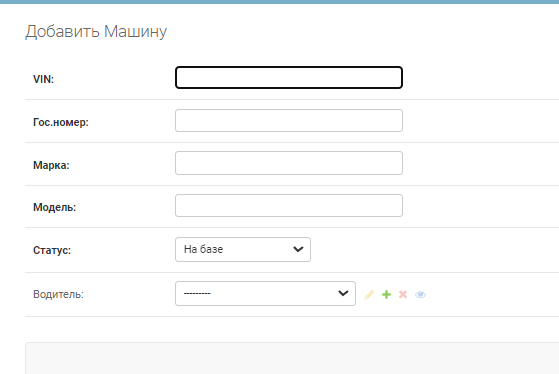


Рисунок 4.2.18 – Страница с формой создания машины

При нажатии на кнопку «Добавить» в админ-панели, система выводит форму для создания склада (рисунок 4.2.19).

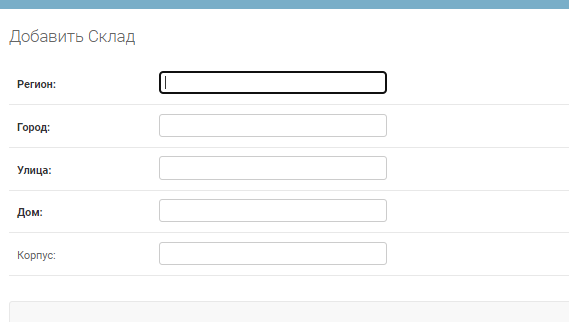


Рисунок 4.2.19 – Страница с формой создания склада

При нажатии на кнопку «Изменить» в админ-панели, система выводит форму для изменения склада (рисунок 4.2.20).

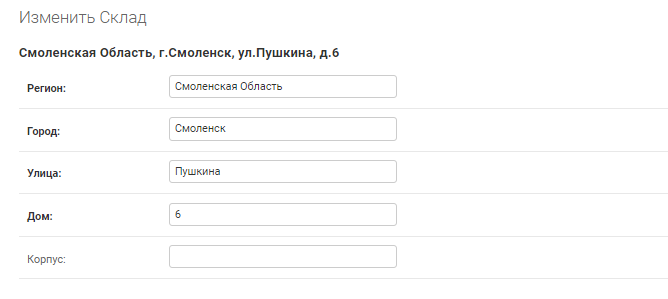


Рисунок 4.2.20 – Страница с формой изменения склада

При нажатии на кнопку «Изменить» в админ-панели, система выводит форму для изменения пункта выдачи (рисунок 4.2.21).

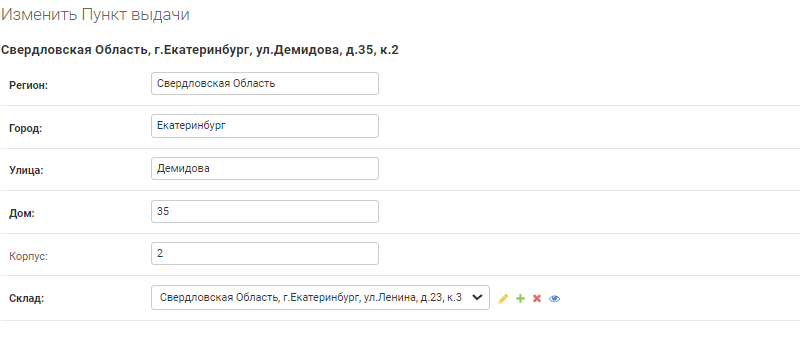


Рисунок 4.2.21 – Страница с формой изменения пункта выдачи

Заключение

В заключении курсового проекта можно подчеркнуть ключевые моменты, представленные в предыдущих разделах. Проект по разработке программного обеспечения для логистической компании "Быстрее света" выявил актуальность автоматизации логистических процессов в современной бизнес-среде. Развитие информационных решений в данной области становится важным фактором для повышения конкурентоспособности и эффективности логистических компаний.

Определенные в постановке задачи цели, связанные с созданием программного продукта, направленного на учет и управление посылками, были детально раскрыты в системном проекте. Описание предметной области, логическая и физическая структуры базы данных, а также описание основных этапов работы логистической компании были представлены для обеспечения полного понимания процессов, подлежащих автоматизации.

Проанализированы сфера деятельности компании, её взаимодействие с клиентами, сотрудниками и внутренними ресурсами. Выделены ключевые моменты, такие как процессы отправки и получения посылок, расчёт стоимости доставки, уведомление клиентов о сроках и последующая обработка не забранных посылок.

Реализация поставленных задач включала в себя выбор программных и технических средств, разработку базы данных, написание программного обеспечения и проведение тестирования. Описаны ключевые шаги от проектирования до внедрения программного продукта.

Этот курсовой проект продемонстрировал не только теоретические аспекты разработки программного обеспечения, но и практическое применение современных технологий в логистической индустрии. Полученный опыт и результаты могут служить основой для дальнейших исследований и разработок в области логистики и информационных технологий.

Список использованных источников

1. 1. Сайт Django Project [Электронный ресурс]. – Режим доступа: https://docs.djangoproject.com/en/5.0/topics/db/models/
2. Сайт Bootstrap [Электронный ресурс]. – Режим доступа: <https://getbootstrap.com/docs/5.3/getting-started/introduction/;>
3. Сайт freecodecamp [Электронный ресурс]. – Режим доступа:  [https://www.freecodecamp.org/news/what-is-one-to-many-relationship-in-django/;](https://studfile.net/preview/5910972/page:6/;)
4. Сайт METANIT [Электронный ресурс]. – Режим доступа: <https://metanit.com/python/django/5.9.php;>
5. Сайт GitBook [Электронный ресурс]. – Режим доступа: [https://ilyachch.gitbook.io/django-rest-framework-russian-documentation/overview/readme;](https://metanit.com/python/django/5.9.php;)