

Министерство науки и высшего образования РФ
Федеральное государственное автономное
образовательное учреждение высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий
Кафедра вычислительной техники

КУРСОВОЙ ПРОЕКТ

Информационная система «Грузоперевозки»

Руководитель

подпись, дата

В. С. Васильев

инициалы, фамилия

Студент

КИ22-06Б, 032215219

номер группы, зачётной книжки

подпись, дата

А. В. Кулешов

инициалы, фамилия

Красноярск 2024

РЕФЕРАТ

Курсовой проект состоит из 25 страниц текста, 000 таблиц, 000 рисунков, 000 использованных источников и 1 приложения.

ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ, ИНФОРМАЦИОННАЯ СИСТЕМА, ICONIX, ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ Целью проекта является проектирование и реализация информационной системы «Грузоперевозки» согласно полученному техническому заданию.

В первой главе приведено описание выделенных ролей в системе, их прецедентов. Описаны форматы хранимых и обрабатываемых данных.

Во второй главе показаны результаты, полученные на этапе объектно-ориентированного проектирования системы. Приведены диаграммы пригодности и последовательности, диаграммы классов и ER-диаграмма.

В третьей главе описаны особенности программной реализации системы, а также инструкция по сборке и первому запуску программы.

					<i>КП 09.03.01 – 032215219 – ПЗ</i>		
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>			
<i>Разраб.</i>		<i>Кулешов А. В.</i>			<i>Информационная система «Грузоперевозки»</i>		
<i>Пров.</i>		<i>Васильев В. С.</i>					
<i>Н. контр.</i>							
<i>Утв.</i>							
						<i>Лит.</i>	<i>Лист</i>
							<i>Листов</i>
							2129
						<i>КИ22-06Б</i>	

СОДЕРЖАНИЕ

Введение	4
1 Разработка спецификации требований	5
1.1 Выявление ролей и функций, диаграммы прецедентов.....	5
1.2 Текстовое описание прецедентов и макеты интерфейса.....	5
1.2.1 Прецедент «войти в систему».....	5
1.2.2 Прецедент «сменить пароль».....	7
1.2.3 Прецедент «выйти из системы».....	7
1.2.4 Прецедент «просмотреть зарплату водителя».....	7
1.2.5 Прецедент «просмотреть список пользователей».....	8
1.2.6 Прецедент «назначить перевозку»	8
1.2.7 Прецедент «отменить перевозку»	8
1.2.8 Прецедент «завершить перевозку».....	9
1.2.9 Прецедент «добавить маршрут»	9
1.2.10 Прецедент «удалить маршрут»	10
1.2.11 Прецедент «посмотреть свои перевозки»	10
1.3 Описание форматов данных	10
2 Объектно-ориентированное проектирование.....	11
2.1 Диаграммы пригодности и последовательности.....	11
2.1.1 Прецедент «войти в систему».....	11
2.1.2 Прецедент «сменить пароль».....	12
2.1.3 Прецедент «назначить перевозку»	12
2.1.4 Прецедент «просмотреть список пользователей».....	12
2.1.5 Прецедент «просмотреть свои перевозки».....	12
2.2 ER-диаграмма	12
2.3 Диаграмма классов.....	13
3 Объектно-ориентированное программирование	21
3.1 Реализация.....	21
3.2 Сборка и запуск.....	21
3.3 Тестирование	22
3.4 Инструкции.....	22
Заключение	23
Список использованных источников	24
Приложение А Диаграммы потоков экранов.....	24

ВВЕДЕНИЕ

Целью работы является разработка приложения «Грузоперевозки» с открытым исходным кодом, позволяющего автоматизировать операции: заключения договоров, расчёта заработной платы, регистрации новых сотрудников и пр.

					<i>КП 09.03.01 – 032215219 – ПЗ</i>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		4

1 Разработка спецификации требований

1.1 Выявление ролей и функций, диаграммы прецедентов

В разрабатываемой системе были выделены следующие роли: водитель, логист, бухгалтер. От введения отдельной роли администратора было решено отказаться, поскольку для любых его задач значительно эффективнее будет работать напрямую с базой данных, применяя специализированные инструменты.

На рисунках 1, 2 и 3 представлены диаграммы прецедентов для выделенных ролей. На рисунке 4 представлена диаграмма прецедентов, которые относятся сразу ко всем ролям.

В приложении А представлены диаграммы потоков экранов.

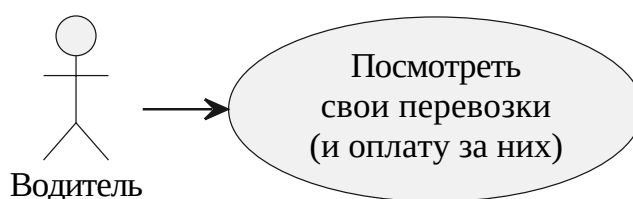


Рисунок 1 – Диаграмма прецедентов водителя

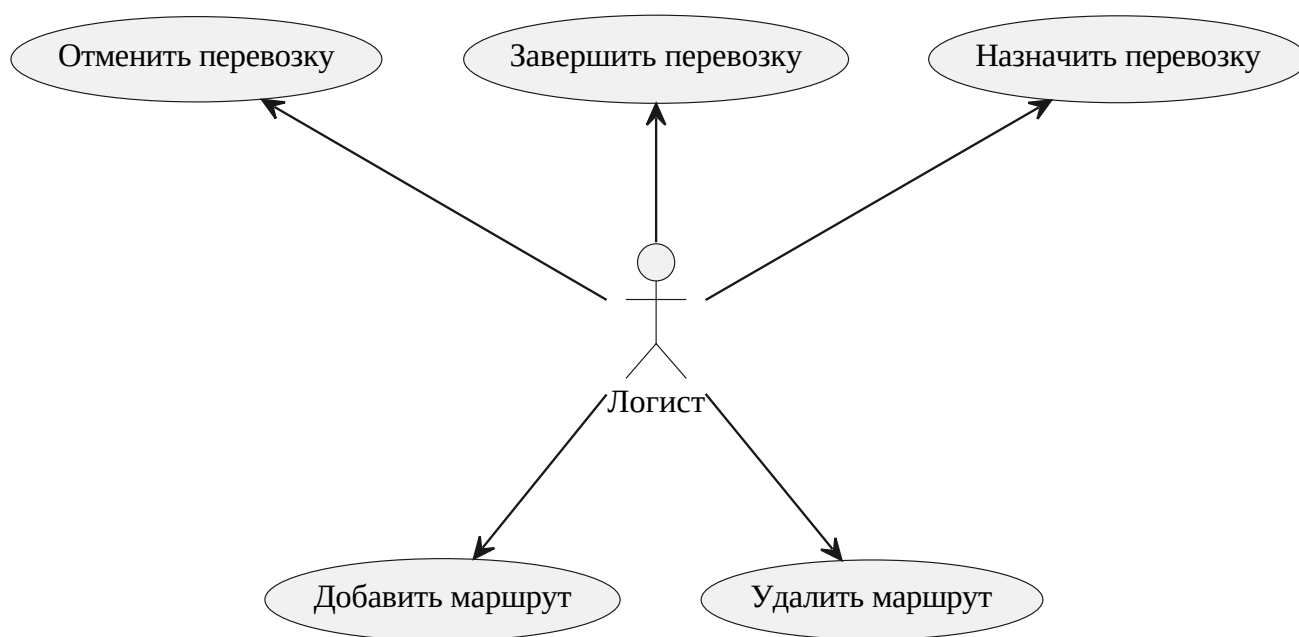


Рисунок 2 – Диаграмма прецедентов логиста

1.2 Текстовое описание прецедентов и макеты интерфейса

1.2.1 Прецедент «войти в систему»

Роли: все

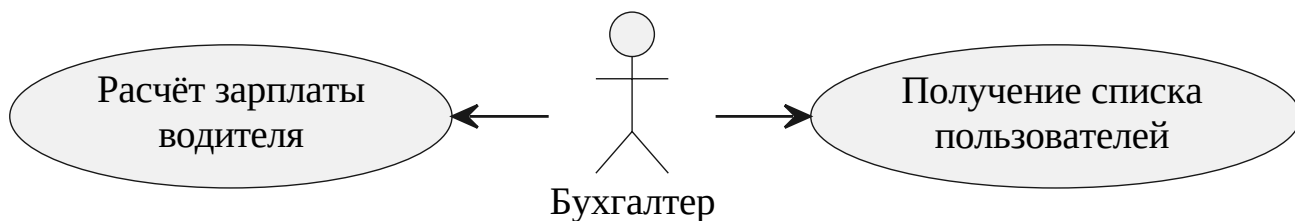


Рисунок 3 – Диаграмма прецедентов бухгалтера

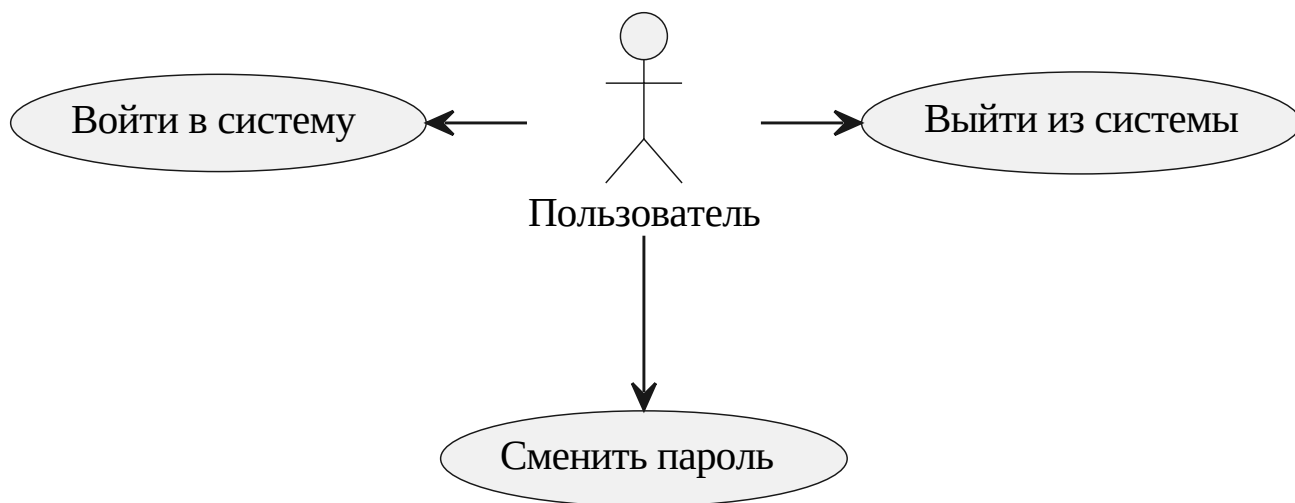


Рисунок 4 – Диаграмма прецедентов всех пользователей

Цель сценария: войти в систему

Предусловия: открыто «Окно входа в программу» (рис. ??)

Основной сценарий:

1. Ввести имя пользователя;
2. Ввести пароль;
3. Нажать кнопку «Войти»;
4. Система проверяет наличие пользователя в базе;
5. Система проверяет, что введён правильный пароль;
6. Пользователь существует и пароль введён верно, поэтому система открывает «Домашнее окно».

Постусловия: открыто «Домашнее окно», окно входа закрыто.

Альтернативная последовательность (пользователь не существует):

1. Вся последовательность сохраняется. Однако указанный пользователь отсутствует в базе.

Постусловия: отображается сообщение об ошибке.

Альтернативная последовательность (пароль не верен):

1. Вся последовательность сохраняется. Однако введён неверный пароль.

Постусловия: отображается сообщение об ошибке.

1.2.2 Прецедент «сменить пароль»

Роли: все

Цель сценария: изменить пароль пользователя для входа в систему

Предусловия: открыто окно «Профиль» (рис. ??)

Основной сценарий:

1. Ввести текущий пароль в поле «Текущий пароль»;
2. Ввести новый пароль в поле «Новый пароль»;
3. Ввести новый пароль в поле «Повторите новый пароль»;
4. Нажать кнопку «Изменить пароль»;
5. Система проверяет совпадение пароля в двух полях ввода;
6. Пароли совпадают, поэтому выводится окно с запросом подтверждения;
7. В появившемся окне (рис. 10) нажать кнопку «Подтвердить».

Постусловия: пароль пользователя для входа в систему изменён.

Альтернативная последовательность (отказ от смены пароля):

1. Последовательность сохраняется до нажатия кнопки «Подтвердить», вместо которой нажата кнопка «Отмена».

Постусловия: отображается сообщение об ошибке.

Альтернативная последовательность (пароли не совпадают):

1. Вся последовательность сохраняется. Однако пароли не совпадают.

Постусловия: отображается сообщение об ошибке.

1.2.3 Прецедент «выйти из системы»

Роли: все

Цель сценария: выйти из системы

Предусловия: открыто окно «Профиль» (рис. ??)

Основной сценарий:

1. Нажать кнопку «Выйти из системы»;
2. В появившемся окне (рис. ??) нажать кнопку «Подтвердить».

Постусловия: сеанс пользователя будет завершён, ранее открытые окна закрыты, откроется «Окно входа в программу»

Альтернативная последовательность (остаться в системе):

1. Последовательность сохраняется до нажатия кнопки «Подтвердить», вместо которой нажата кнопка «Отмена».

Постусловия: Возврат к домашнему окну

1.2.4 Прецедент «просмотреть зарплату водителя»

Роли: бухгалтер

Цель сценария: получить список завершённых перевозок водителя с информацией о вознаграждении за них

Предусловия: открыта вкладка «Водители» в домашнем окне (рис. ??), список водителей не пуст

					<i>КП 09.03.01 – 032215219 – ПЗ</i>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		7

Основной сценарий:

1. Выбрать в списке нужного водителя;
2. Нажать кнопку «Перевозки»;
3. В появившемся окне (рис. ??) выбрать один из предусмотренных периодов либо задать желаемый период вручную.

Постусловия: доступен к просмотру список завершённых перевозок водителя с информацией о вознаграждении за них

1.2.5 Прецедент «просмотреть список пользователей»

Роли: бухгалтер

Цель сценария: получить список пользователей с заданной ролью

Предусловия: открыто домашнее окно

Основной сценарий:

1. Переключиться на вкладку с требуемой ролью пользователей

Постусловия: доступен к просмотру список пользователей с выбранной ролью, содержащий всю имеющуюся в базе данных информацию

1.2.6 Прецедент «назначить перевозку»

Роли: логист

Цель сценария: назначить новую перевозку

Предусловия: открыта вкладка «Перевозки» в домашнем окне

Основной сценарий:

1. Нажать кнопку «Добавить»;
2. В появившемся окне (рис. ??) выбрать нужного водителя в выпадающем списке «Водитель»;
3. Если нужен второй водитель, выбрать его в выпадающем списке «Второй водитель»;
4. Выбрать маршрут в выпадающем списке «Маршрут»;
5. Если нужно, ввести значения премий водителей за данную перевозку;
6. Нажать кнопку «Подтвердить».

Постусловия: назначена новая перевозка

Альтернативная последовательность (возврат к списку перевозок без назначения перевозки):

1. Последовательность сохраняется до нажатия кнопки «Подтвердить», вместо которой нажата кнопка «Отмена».

Постусловия: возврат к домашнему окну, изменения не вносятся

1.2.7 Прецедент «отменить перевозку»

Роли: логист

Цель сценария: пометить перевозку отменённой

Предусловия: открыта вкладка «Перевозки» в домашнем окне, список перевозок не пуст

Основной сценарий:

1. Выбрать в списке перевозку, которую необходимо отменить;
2. Нажать кнопку «Отменить»;
3. В появившемся окне нажать кнопку «Подтвердить».

Постусловия: перевозка помечена отменённой

Альтернативная последовательность (возврат к списку перевозок без отмены перевозки):

1. Последовательность сохраняется до нажатия кнопки «Подтвердить», вместо которой нажата кнопка «Отмена».

Постусловия: возврат к домашнему окну, изменения не вносятся

1.2.8 Прецедент «завершить перевозку»

Роли: логист

Цель сценария: пометить перевозку успешно завершённой

Предусловия: открыта вкладка «Перевозки» в домашнем окне, список перевозок не пуст

Основной сценарий:

1. Выбрать в списке перевозку, которую необходимо завершить;
2. Нажать кнопку «Завершить»;
3. В появившемся окне нажать кнопку «Подтвердить».

Постусловия: перевозка помечена успешно завершённой

Альтернативная последовательность (возврат к списку перевозок без завершения перевозки):

1. Последовательность сохраняется до нажатия кнопки «Подтвердить», вместо которой нажата кнопка «Отмена».

Постусловия: возврат к домашнему окну, изменения не вносятся

1.2.9 Прецедент «добавить маршрут»

Роли: логист

Цель сценария: внести в базу новый маршрут

Предусловия: открыта вкладка «Маршруты» в домашнем окне (рис. ??)

Основной сценарий:

1. Нажать кнопку «Добавить»;
2. В появившемся окне (рис. ??) ввести данные маршрута;
3. Нажать кнопку «Подтвердить».

Постусловия:

Альтернативная последовательность ():

Постусловия: в базу добавлен новый маршрут

Альтернативная последовательность (возврат к списку маршрутов без добавления нового маршрута):

					<i>КП 09.03.01 – 032215219 – ПЗ</i>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		9

1. Последовательность сохраняется до нажатия кнопки «Подтвердить», вместо которой нажата кнопка «Отмена».

Постусловия: возврат к домашнему окну, изменения не вносятся

1.2.10 Прецедент «удалить маршрут»

Роли: логист

Цель сценария: пометить маршрут в базе неиспользуемым

Предусловия: открыта вкладка «Маршруты» в домашнем окне, список маршрутов не пуст

Основной сценарий:

1. Нажать кнопку «Удалить»;
2. В появившемся окне (рис. ??) ввести данные маршрута;
3. Нажать кнопку «Подтвердить».

Постусловия:

Альтернативная последовательность (0):

Постусловия: маршрут в базе помечен, как более не используемый

Альтернативная последовательность (возврат к списку маршрутов без удаления маршрута):

1. Последовательность сохраняется до нажатия кнопки «Подтвердить», вместо которой нажата кнопка «Отмена».

Постусловия: возврат к домашнему окну, изменения не вносятся

1.2.11 Прецедент «посмотреть свои перевозки»

Роли: водитель

Цель сценария: просмотреть список перевозок и зарплату за определённый период

Предусловия: открыта вкладка «Перевозки» в домашнем окне (рис. ??)

Основной сценарий:

1. В области редактирования периодов выбрать один из предусмотренных периодов либо задать желаемый период вручную;
2. Система проверяет валидность периода;
3. Период валиден, поэтому система отображает список завершённых в выбранный период перевозок с информацией об оплате за них по отдельности и в сумме за все.

Постусловия: доступен к просмотру список завершённых в выбранный период перевозок с информацией об оплате за них

Альтернативная последовательность (указан невалидный период):

1. Вся последовательность сохраняется. Однако период невалиден.

Постусловия: отображается сообщение об ошибке

1.3 Описание форматов данных

					<i>КП 09.03.01 – 032215219 – ПЗ</i>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		10

2 Объектно-ориентированное проектирование

2.1 Диаграммы пригодности и последовательности

2.1.1 Прецедент «войти в систему»

На рисунке 5 представлена диаграмма пригодности для данного прецедента. На рисунке 6 представлена диаграмма последовательности для данного прецедента.

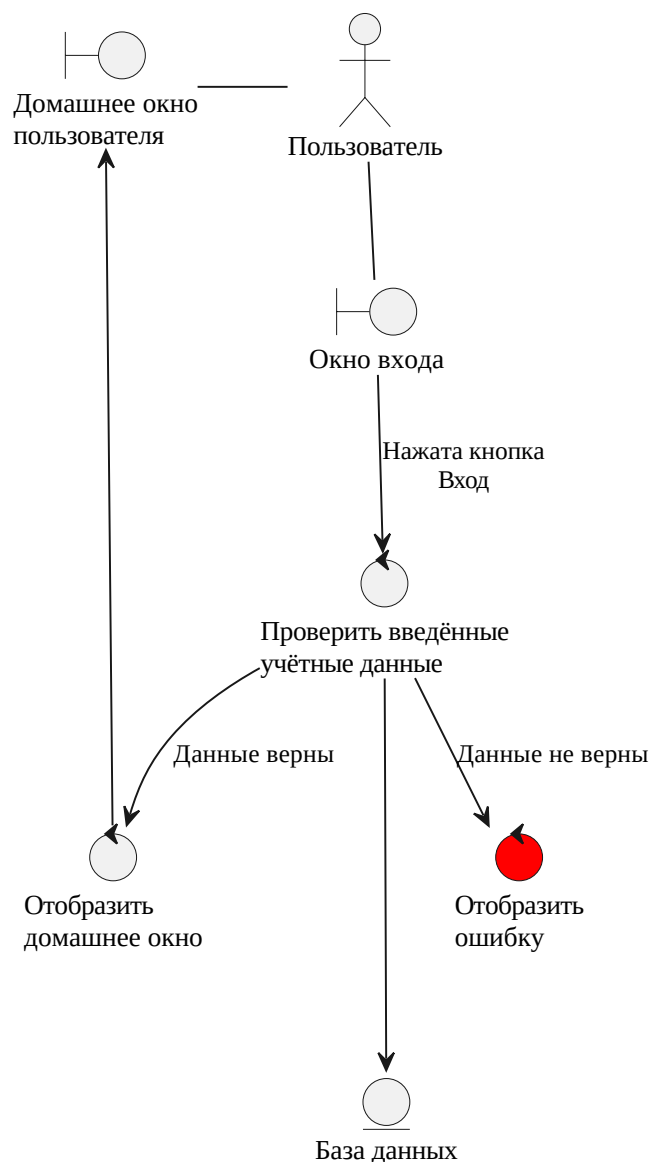


Рисунок 5 – Диаграмма пригодности для прецедента «войти в систему»

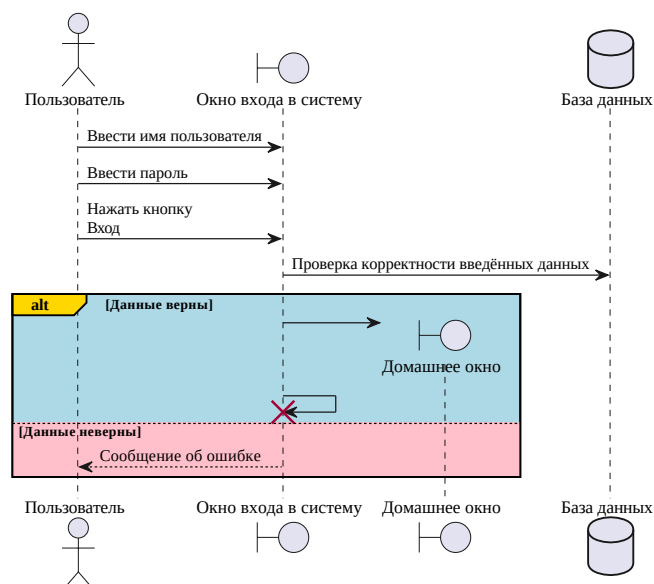


Рисунок 6 – Диаграмма последовательности для прецедента «войти в систему»

2.1.2 Прецедент «сменить пароль»

На рисунке 7 представлена диаграмма пригодности для данного прецедента. На рисунке 8 представлена диаграмма последовательности для данного прецедента.

2.1.3 Прецедент «назначить перевозку»

На рисунке 9 представлена диаграмма пригодности для данного прецедента. На рисунке 10 представлена диаграмма последовательности для данного прецедента.

2.1.4 Прецедент «просмотреть список пользователей»

На рисунке 11 представлена диаграмма пригодности для данного прецедента. На рисунке 12 представлена диаграмма последовательности для данного прецедента.

2.1.5 Прецедент «просмотреть свои перевозки»

На рисунке 13 представлена диаграмма пригодности для данного прецедента. На рисунке 14 представлена диаграмма последовательности для данного прецедента.

2.2 ER-диаграмма

На рисунке 15 представлена ER-диаграмма разрабатываемого приложения в нотации Мартина.

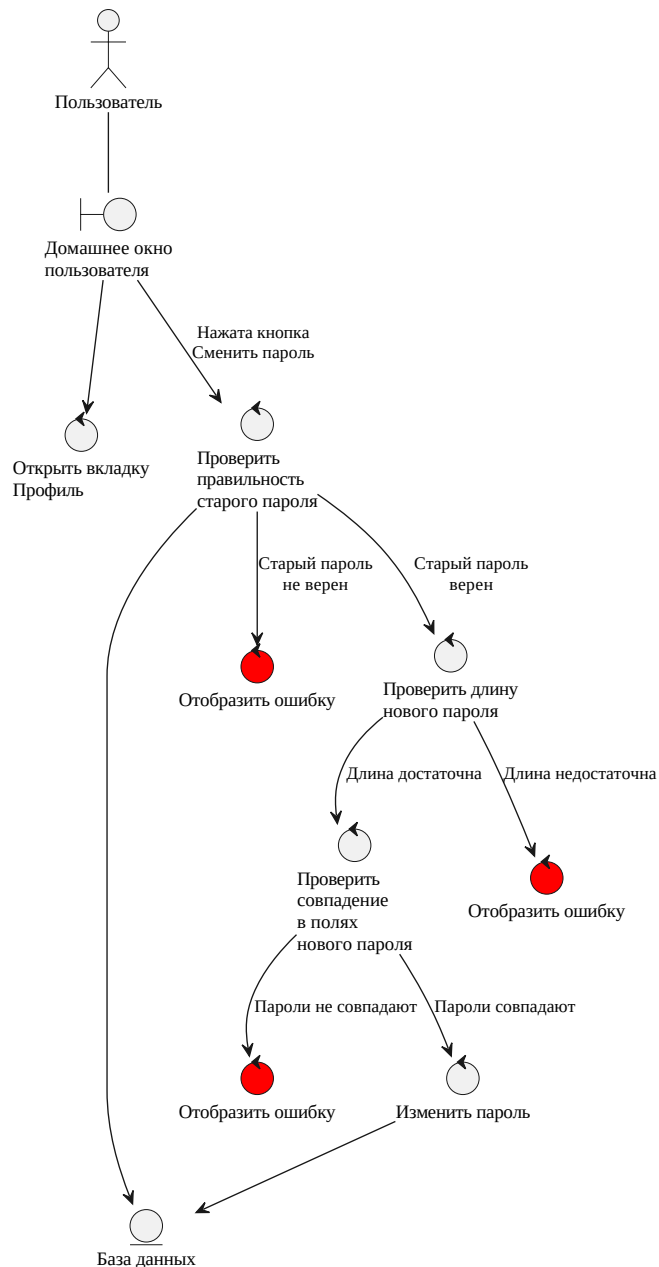


Рисунок 7 – Диаграмма пригодности для прецедента «сменить пароль»

2.3 Диаграмма классов

На рисунке 16 представлена диаграмма классов уровня проектирования для разрабатываемого приложения.

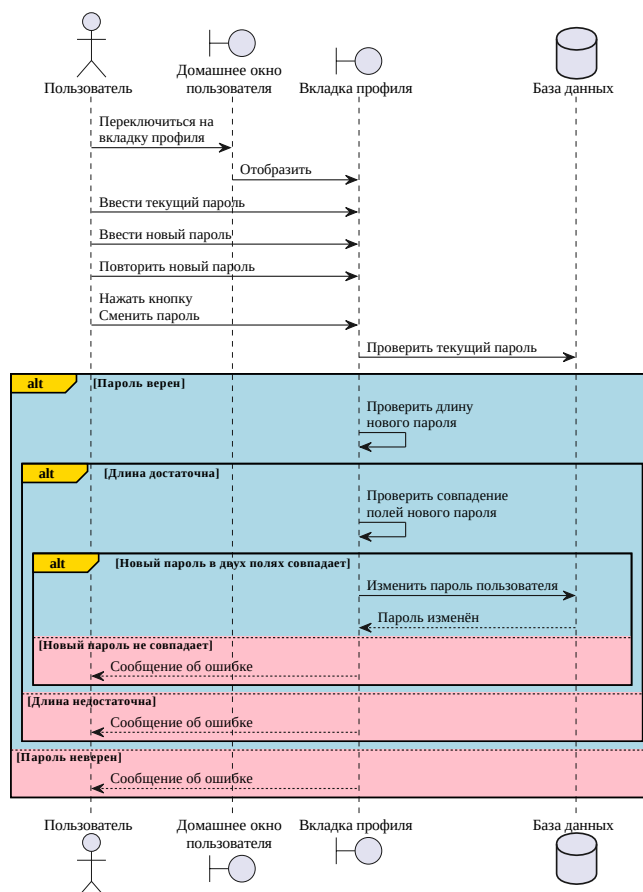


Рисунок 8 – Диаграмма последовательности для прецедента «сменить пароль»

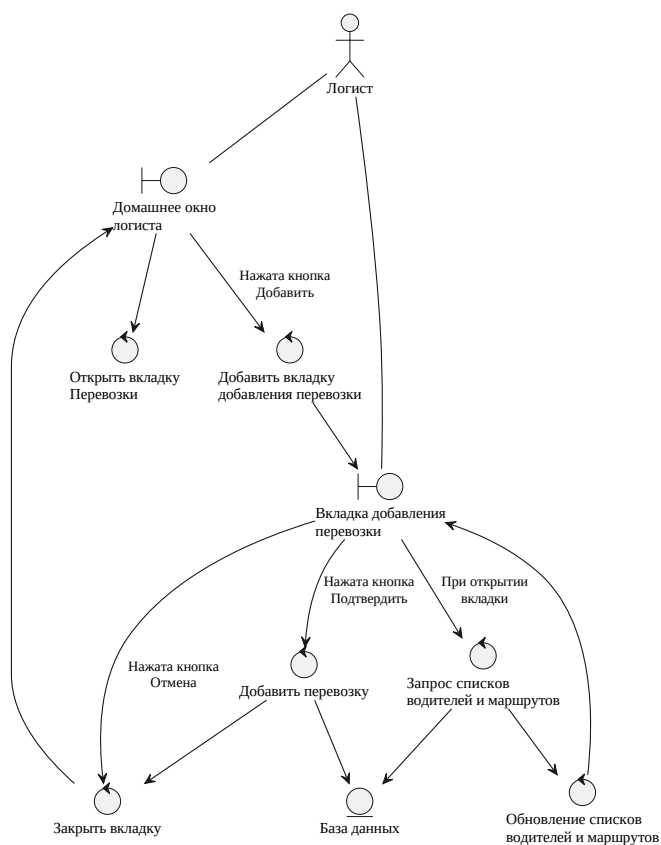


Рисунок 9 – Диаграмма пригодности для прецедента «назначить перевозку»

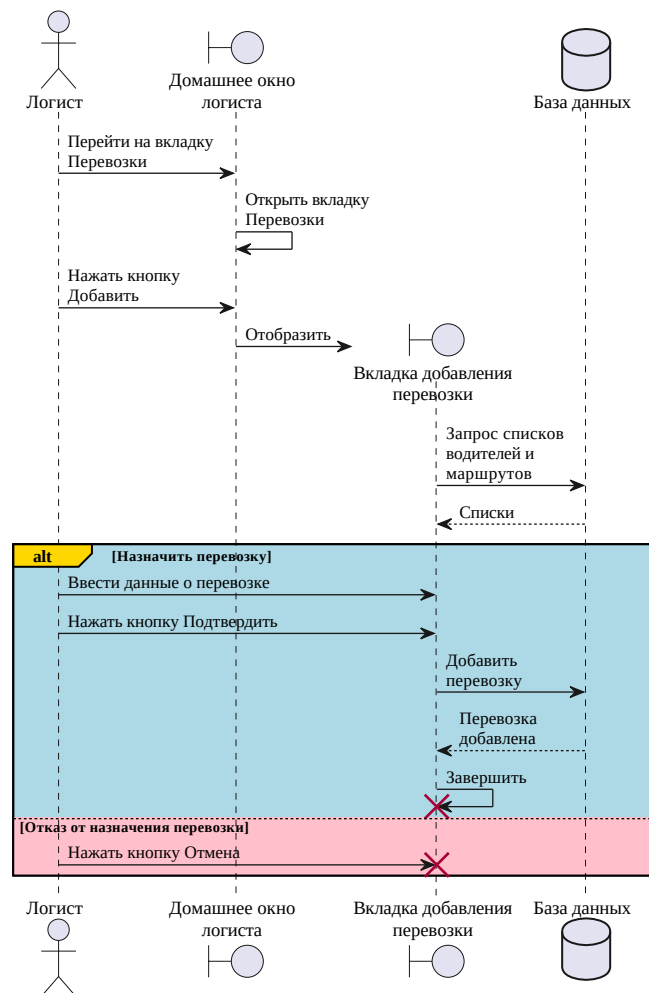


Рисунок 10 – Диаграмма последовательности для прецедента «назначить перевозку»

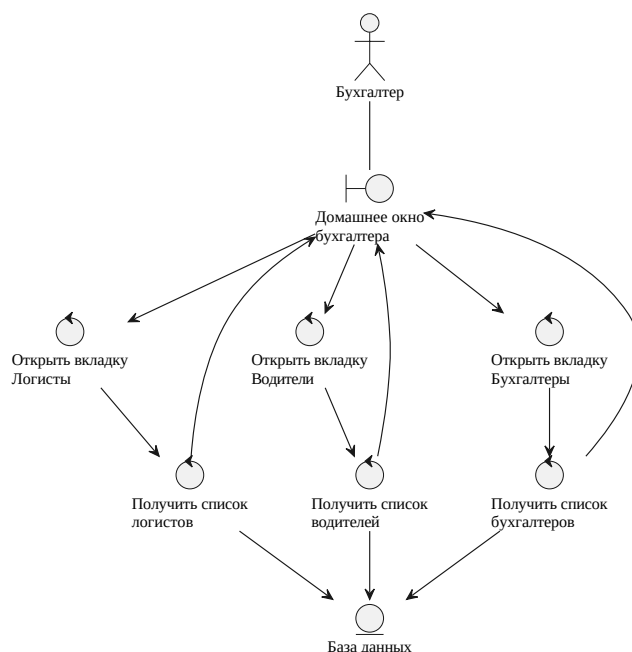


Рисунок 11 – Диаграмма пригодности для прецедента «просмотреть список пользователей»

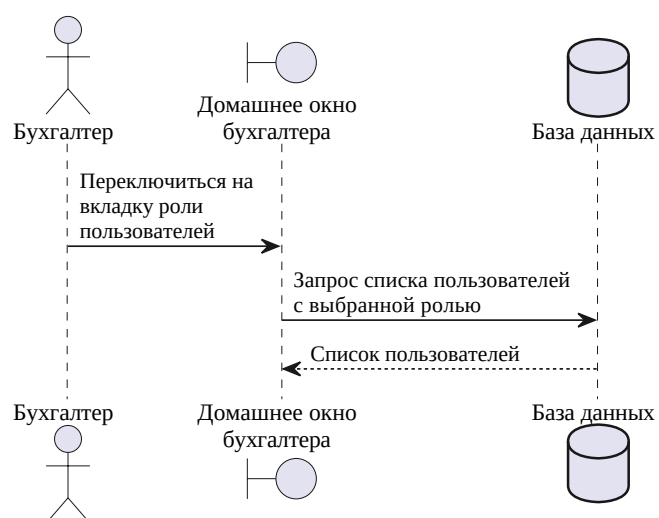


Рисунок 12 – Диаграмма последовательности для прецедента «просмотреть список пользователей»

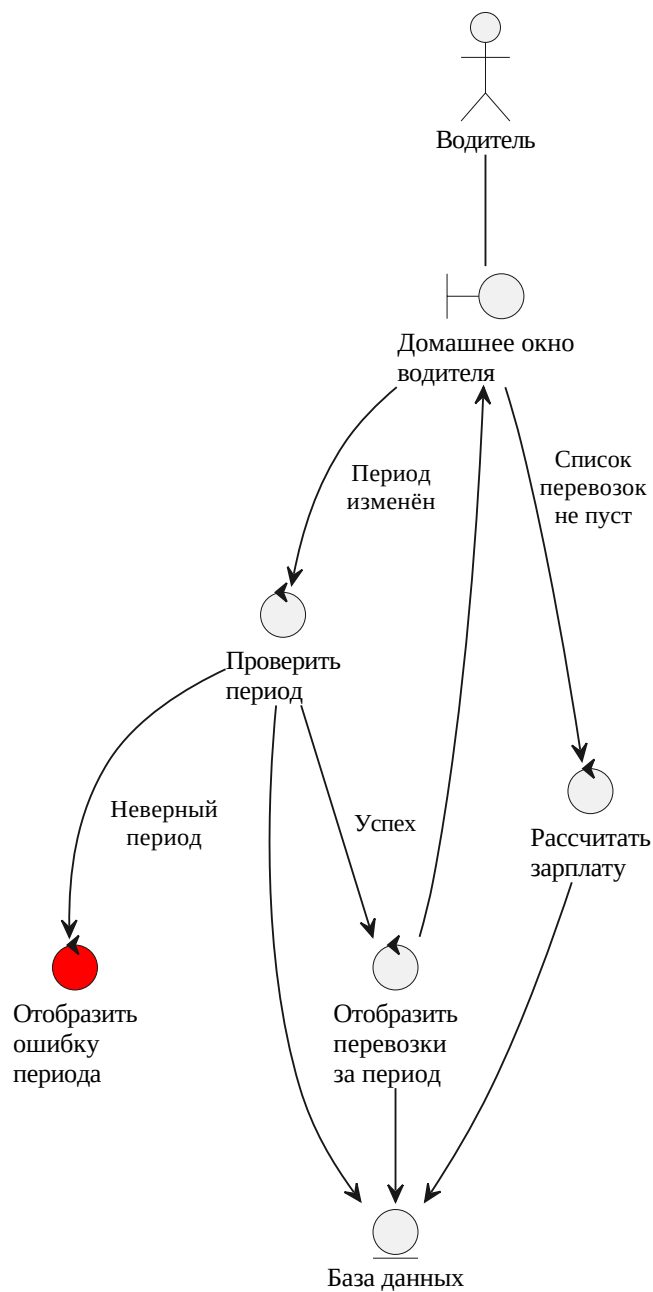


Рисунок 13 – Диаграмма пригодности для прецедента «просмотреть свои перевозки»

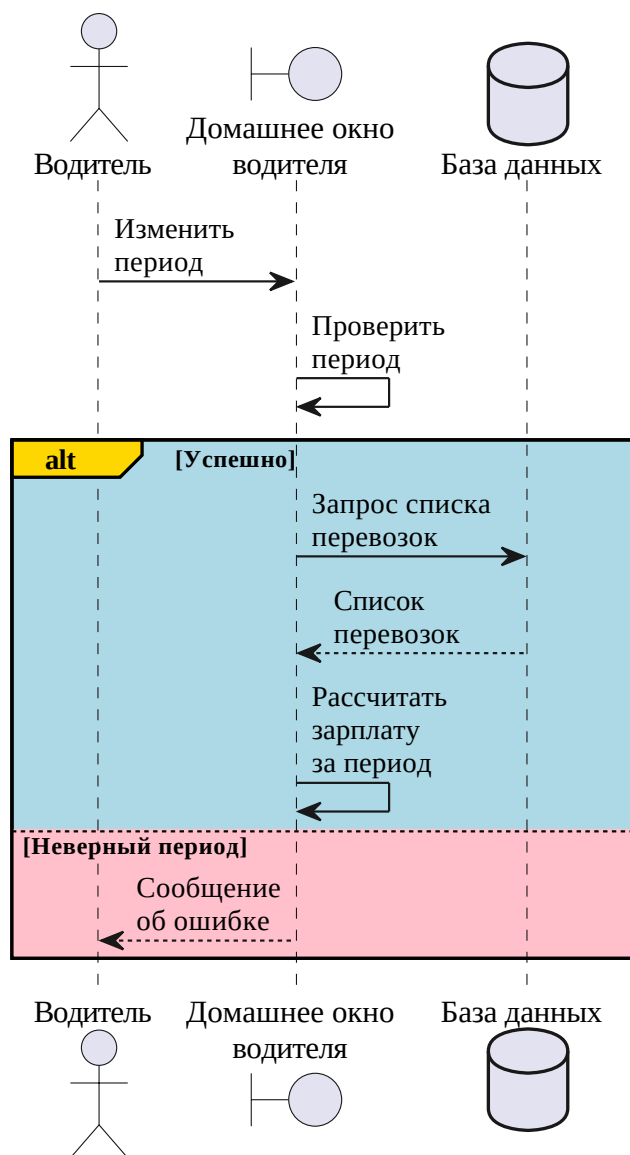


Рисунок 14 – Диаграмма последовательности для прецедента «просмотреть свои перевозки»

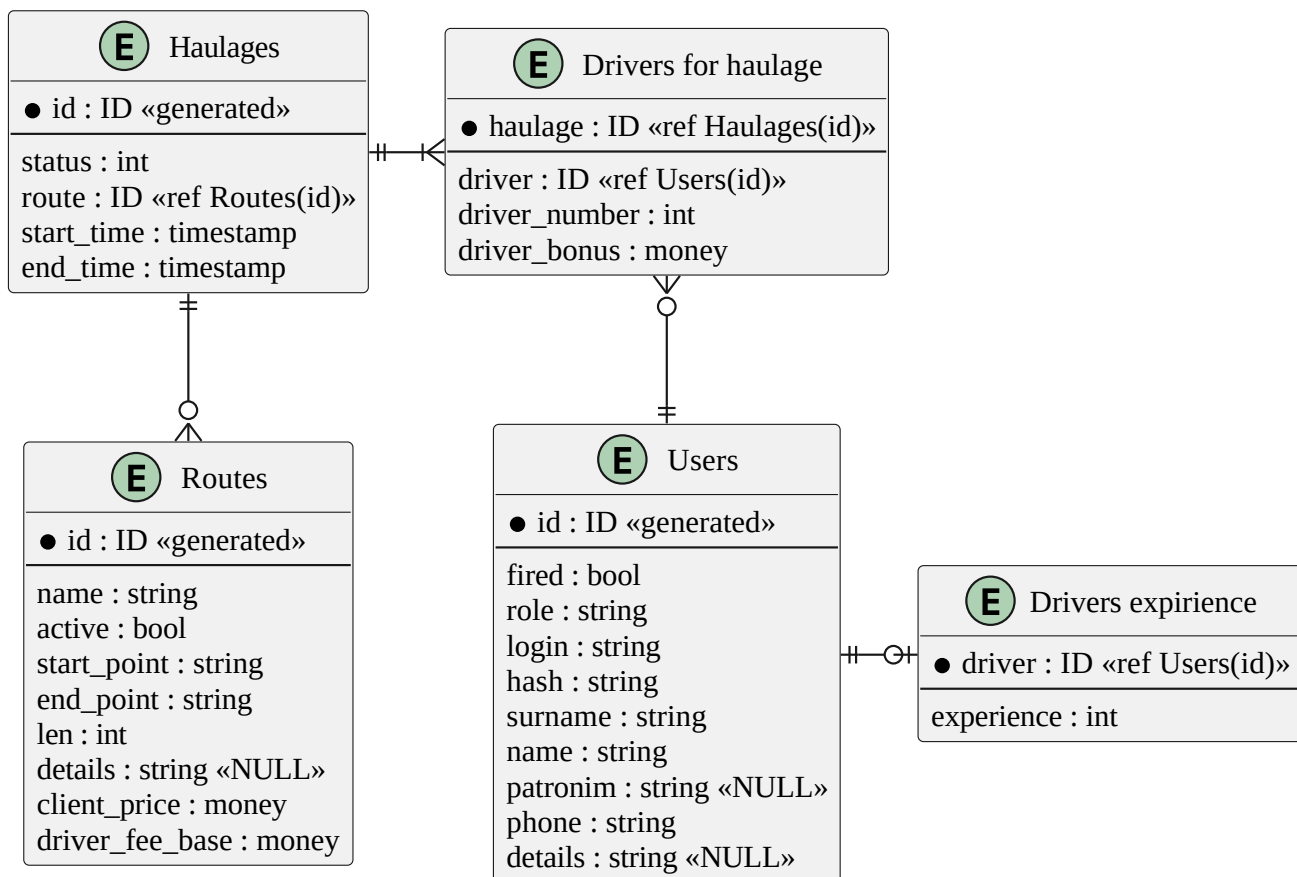


Рисунок 15 – ER-Диаграмма

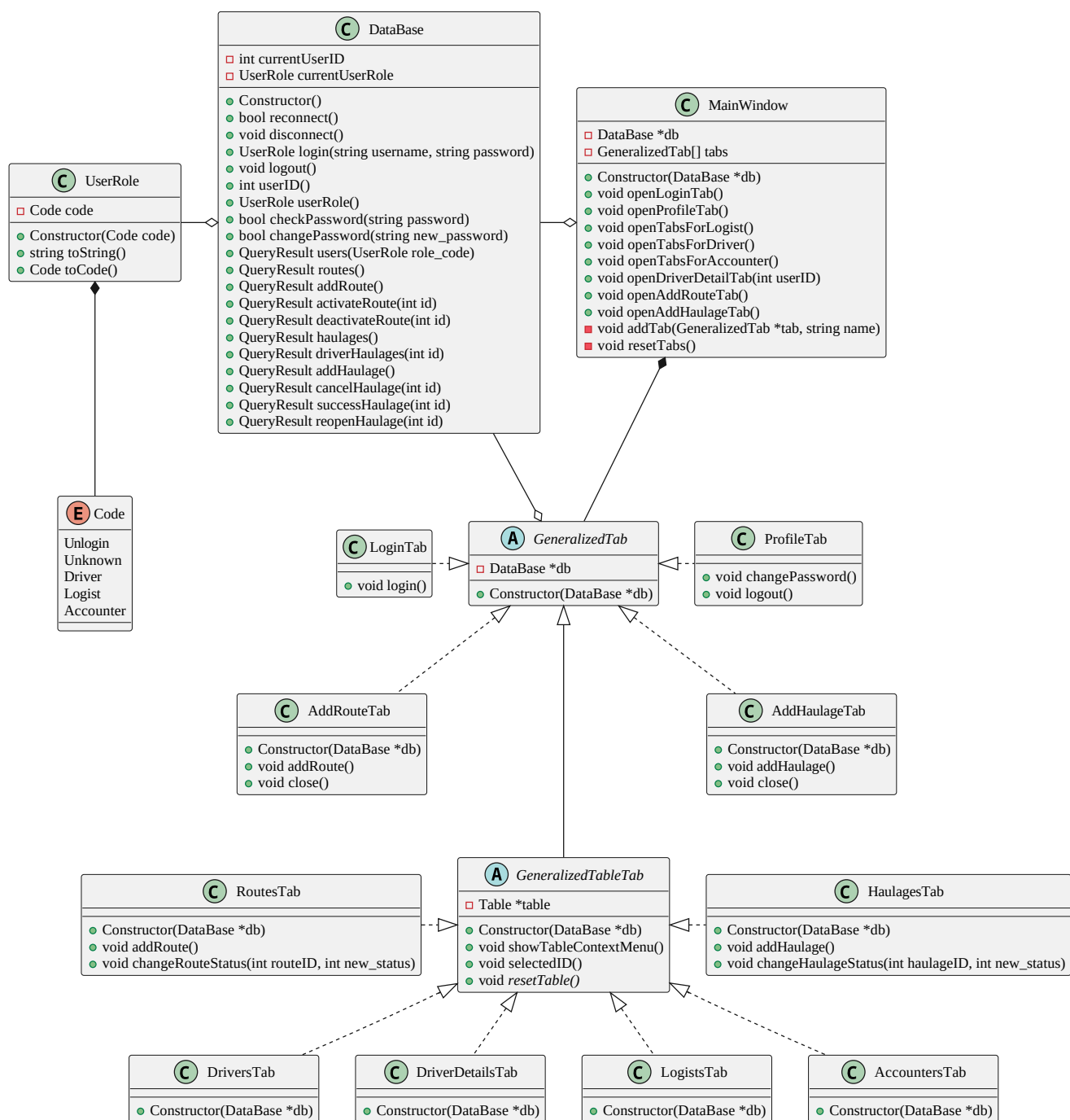


Рисунок 16 – Диаграмма классов уровня проектирования

3 Объектно-ориентированное программирование

3.1 Реализация

Информационная система была реализована на языке программирования C++ с использованием фреймворка Qt версии 6.7.1. Для хранения и обработки данных информационной системы была выбрана клиент-серверная СУБД PostgreSQL.

Классы, ответственные за формирование и выполнение запросов к базе данных, используют параметризованные SQL запросы для предотвращения SQL-инъекций.

Ни одна запись в таблице не может быть безвозвратно удалена из интерфейса программы, для чего применён паттерн Soft Delete с помощью добавления дополнительного поля либо состояния существующего поля к записям, для которых предусмотрена возможность удаления.

Для организации взаимодействия пользователя с системой была выбрана архитектура Model-View, встроенная в Qt и являющаяся вариацией паттерна Model-View-Controller. В качестве модели (Model) используется QSqlQueryModel. В качестве представления (View) используется QTableView.

Исходный код программы представлен в свободном доступе на GitHub.

3.2 Сборка и запуск

Сборку и запуск программы можно выполнить через IDE Qt Creator, выбрав файл проекта `gruzop.pro` и запустив сборку через графический интерфейс.

Также сборку можно произвести в ручном режиме. Ниже описанные действия выполняются в директории с исходными текстами программы. Программа предназначена для ОС семейства Linux. Для сборки программы необходимо выполнить следующие команды:

```
$ qmake path\to\gruzop.pro
$ make -j8
$ make clean
```

Параметр `-j8` не является обязательным и предназначен для многопоточной сборки, что может ускорить процесс сборки, в данном случае будет задействовано 8 потоков сборки. Для изменения числа потоков замените 8 на требуемое число.

Для работы собранной программы требуется наличие в системе установленного пакета Qt версии не ниже 6.7.0, на более ранних версиях работа программы не гарантируется. Также требуется установленный и запущенный сервер PostgreSQL.

Для корректного первого запуска приложения необходимо запустить скрипт развёртывания базы данных, находящийся в каталоге исходных текстов программы:

```
$ psql -U postgres -f dbinit.sql
```

После этого администратор системы может приступить к созданию профилей пользователей, для чего предлагается использовать прямые запросы к базе данных. Соответствующие примеры могут быть найдены в файле `examples.sql`. Необходимо понимание администратором принципов работы реляционных СУБД и минимальные знания языка SQL.

3.3 Тестирование

Проводилось ручное тестирование информационной системы. В качестве примера ниже приведены действия, которые выполнялись при проверке прецедента «войти в систему»:

- Попытка войти, не заполнив поля имени пользователя и пароля;
- Попытка войти, не заполнив только поле имени пользователя;
- Попытка войти, не заполнив только поле пароля;
- Попытка войти, заполнив поля имени пользователя и пароля пробелами;
- Попытка войти, заполнив только поле имени пользователя пробелами;
- Попытка войти, заполнив только поле пароля пробелами;
- Попытка войти с неверным именем пользователя и паролем;
- Попытка войти с правильным именем пользователя, но без пароля;
- Попытка войти с правильным именем пользователя, но неверным паролем;
- Попытка войти с правильным именем пользователя и паролем;

В данном прецеденте все действия, кроме последнего, должны вызвать сообщение об ошибке.

Тестирование проводилось для каждого прецедента, описанного в разделе 1.

В результате проведённого тестирования были выявлены и устранены следующие ошибки:

- первая ошибка;
- вторая ошибка;
- очень прям пиздец ваще охуеть какая длинная-длиннющая-предлинная всеохватная ошибка, прям ошибище среди ошибищ;
- последняя ошибка.

3.4 Инструкции

В качестве инструкций для пользователя можно использовать прецеденты, описанные в разделе 1 и диаграммы последовательности, разработанные в разделе 2.

ЗАКЛЮЧЕНИЕ

В результате выполнения курсового проекта была спроектирована и реализована информационная система, которая полностью соответствует данному техническому заданию. Получен опыт проектирования информационных систем с использованием диаграмм UML, а также опыт работы с инструментами разработки программного обеспечения.

					<i>КП 09.03.01 – 032215219 – ПЗ</i>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		23

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- [1] СТУ 7.5–07–2021. Стандарт университета «Система менеджмента качества. Общие требования к построению, изложению и оформлению документов учебной деятельности».
- [2] Документация Qt // QT | Tools for Each Stage of Software Development Lifecycle. : [сайт]. – URL: <https://doc.qt.io/> (дата обращения 30.05.2024).
- [3] Документация PostgreSQL : [сайт]. – URL: <https://www.postgresql.org/docs/> (дата обращения: 30.05.2024).
- [4] Основы UML – диаграммы использования (use-case) // Блог программиста – программирование и алгоритмы : [сайт]. – URL: <https://pro-prof.com/>. (дата обращения: 20.04.2024).
- [5] Документация PlantUML : [сайт]. – URL: <https://plantuml.com/> (дата обращения: 23.05.2024).

					<i>КП 09.03.01 – 032215219 – ПЗ</i>	Лист
Изм.	Лист	№ докум.	Подп.	Дата		24

ПРИЛОЖЕНИЕ А

Диаграммы потоков экранов

На рисунках А.1, А.2 и А.3 приведены диаграммы потока экранов для водителя, логиста и бухгалтера соответственно.

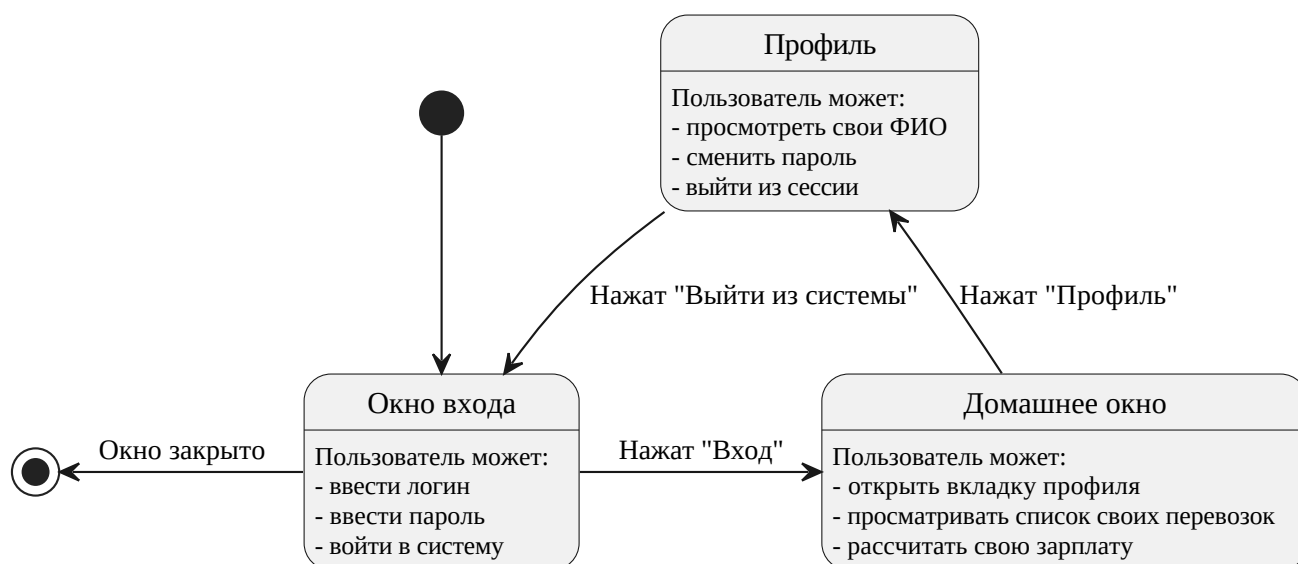


Рисунок А.1 – Диаграмма потока экранов водителя

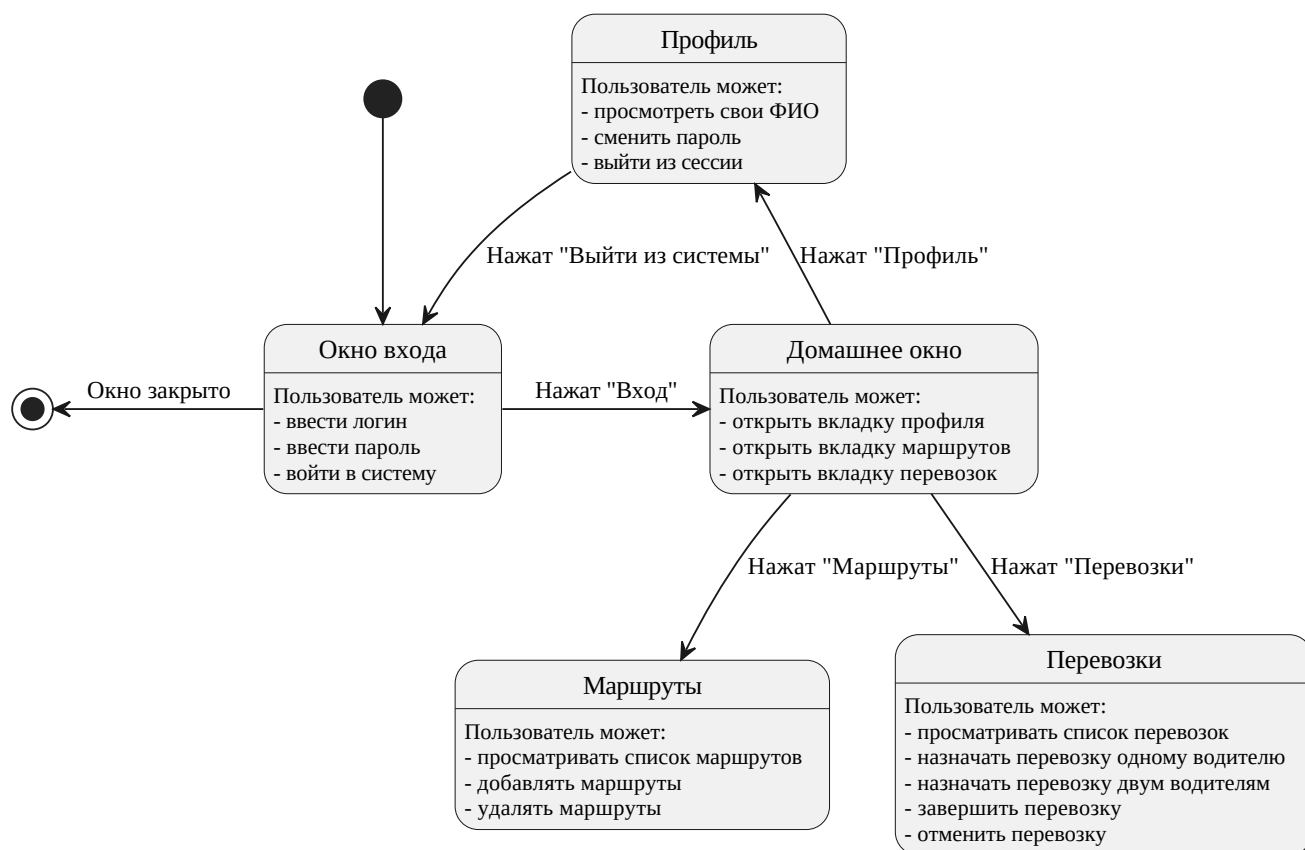


Рисунок А.2 – Диаграмма потока экранов логиста

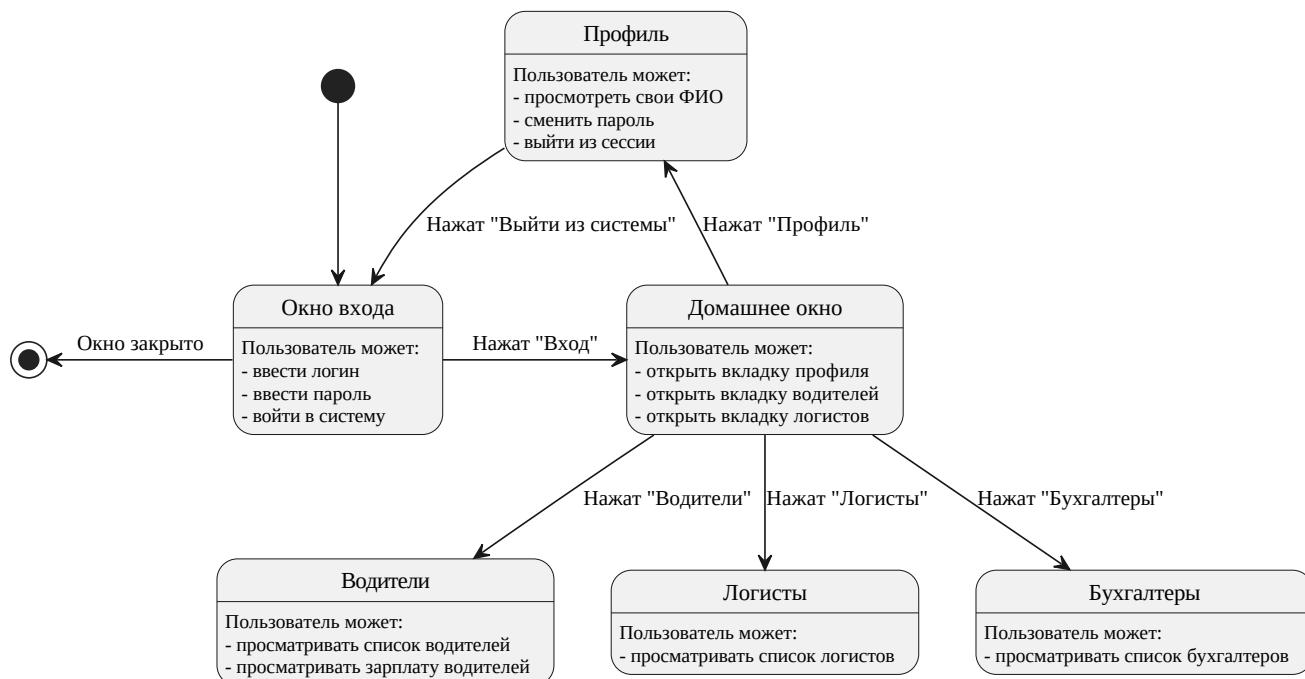


Рисунок А.3 – Диаграмма потока экранов бухгалтера