МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение

высшего образования

**«Дальневосточный федеральный университет»**

**ШКОЛА ЕСТЕСТВЕННЫХ НАУК**

**Кафедра информационных систем управления**

Константинов Остап Владимирович

Разработка базы данных «ГИБДД»

**Курсовой проект**

|  |
| --- |
| Студент гр. Б8319  О.В. Константинов \_\_\_\_\_\_\_\_\_\_\_  подпись |
| Руководитель доцент  Л.В. Красюк \_\_\_\_\_\_\_\_\_\_\_  подпись |
| Регистрационный № \_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  подпись И.О. Фамилия  « \_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2018 г. | Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  подпись И.О. Фамилия  « \_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2018 г. |

г. Владивосток

2018

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение

высшего образования

**«Дальневосточный федеральный университет»**

|  |
| --- |
| **ШКОЛА ЕСТЕСТВЕННЫХ НАУК** |

**Кафедра информационных систем управления**

**Задание**

на курсовой проект

|  |
| --- |
| студенту Константинову О. В. группы Б8319 |

(фамилия, имя, отчество)

на тему Разработка базы данных «ГИБДД»

|  |
| --- |
|  |
|  |

Вопросы, подлежащие разработке (исследованию):

|  |
| --- |
| разработка базы данных; |
| анализ предметной области и ее реализация инструментальными средствами; |
| основы проектирование таблиц, отношение между таблицами; |
| методы проектирования и реализации запросов. |
|  |
|  |
|  |
|  |

Основные источники информации и прочее, используемые для разработки темы

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

Срок представления работы «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2018 г.

Дата выдачи задания «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2018 г.

Оглавление

[Введение 4](#_Toc517264319)

[1. Постановка задачи 5](#_Toc517264320)

[2. Аналитическая часть 7](#_Toc517264321)

[2.1 Описание предметной области 7](#_Toc517264322)

[2.2 Анализ предметной области 7](#_Toc517264323)

[2.2.1 Структура подразделений ГИБДД и их подчиненность 7](#_Toc517264324)

[2.2.2 Бизнес-модель ГИБДД 8](#_Toc517264325)

[2.2.3 Потоки данных ГИБДД 10](#_Toc517264326)

[2.3 Обоснование решаемой задачи 11](#_Toc517264327)

[3. Проектная часть 12](#_Toc517264328)

[3.1 Назначение базы данных 12](#_Toc517264329)

[3.1.1 Описание выходной информации 12](#_Toc517264330)

[3.1.2 Описание входной информации 12](#_Toc517264331)

[3.2 Проектирование базы данных 13](#_Toc517264332)

[3.2.1 Концептуальное проектирование 13](#_Toc517264333)

[3.2.2 Логическое проектирование 16](#_Toc517264334)

[3.2.3 Физическое проектирование 17](#_Toc517264335)

[4. Клиентское приложение 21](#_Toc517264336)

[Заключение 22](#_Toc517264337)

[Список литературы 23](#_Toc517264338)

# Введение

На сегодняшний день применение программных средств получило весьма большое значение для многих организаций, которые для упрощения своей работы применяют компьютерные технологии.

Задачей выполнения курсовой работы является разработка проекта информационной системы и реализация БД для автоматизации работы ГИБДД.

Цель курсовой работы – закрепления знаний, полученных при изучении дисциплины, а также получение практических навыков проектирования и реализации БД с использованием современных технологий и инструментальных средств.

# Постановка задачи

Информация находится в основе любой деятельности. Любая организация имеет в процессе своего функционирования большое количество информации, требующей организации и учёта. Не исключением является и деятельность Государственной инспекции безопасности дорожного движения.

Основу дорожного инспектирования ГИБДД составляет регистрация административных правонарушений. Основная цель протоколирования –выявление и регистрация правонарушения. Анализ составленных протоколов предупреждает значительную часть правонарушений и несчастных случаев, связанных с ними, так как указывает инспекторам на какую именно область правонарушений необходимо обратить особое внимание.

На основании данных протоколирования инспектора заполняют постановление об административном правонарушении, в которых отражается информация о:

* дата и место составления документа;
* должность, фамилия и инициалы инспектора ГИБДД;
* данные о владельце транспортного средства;
* данные о свидетелях и потерпевших, в частности их ФИО и адреса;
* показания свидетелей;
* указание нормативно-правового акта, который был нарушен, место и время совершения правонарушения, а также подробное описание ситуации и роли в ней участников нарушения.
* указание статьи, которая предусматривает ответственность за правонарушение;
* объяснение лица, в отношении которого возбуждено дело;
* иные сведения, которые могут пригодиться для разрешения дела;
* форма административного наказания.

При этом, следует принять во внимание что участниками могут быть как автомобили, так и пешеходы и оба могут фигурировать в постановлении о ДТП в качестве виновников, участников или потерпевших. Вместе с гос. номеров автомобиля требуется хранить информацию о водителе, находящемся за рулем во время ДТП и информацию о владельце автотранспортного средства. Информация о водителе должна опирать на номер паспорта, т. к. у водителя на момент нарушения может не быть водительских прав.

Каждое нарушение ПДД может быть классифицировано по нескольким пунктам КоАП РФ и иметь разные поправочные коэффициента, в зависимости от тяжести правонарушения.

Составление данной отчётности хотя бы за месяц требует пересмотра большого количества документации, что занимает немало времени. Следить за состоянием постановлений по картотеке крайне трудоёмко: данная работа требует больших временных затрат и имеется вероятность совершения большого количества ошибок. Автоматизация данного процесса позволит не только увеличить точность и скорость получения информации, но и как следствие этого позволит инспекторам проверить больше водителей на дороге, а значит увеличить вероятность предотвращения несчастных случаев на дорогах с участием водителей и пешеходов.

# Аналитическая часть

## Описание предметной области

Дорожно-патрульная служба – крупнейшая и наиболее многофункциональная служба в составе Государственной инспекции безопасности дорожного движения МВД России. ДПС – это специфическое направление обеспечения безопасности дорожного движения путем реализации комплекса правовых, технических и естественнонаучных знаний.

Основными задачами дорожно-патрульной службы являются:

* сохранение жизни, здоровья и имущества участников дорожного движения, защита их законных прав и интересов, а также интересов общества и государства;
* обеспечение безопасного и бесперебойного движения транспортных средств;
* предупреждение и пресечение преступлений и административных правонарушений в области дорожного движения.

## Анализ предметной области

### Структура подразделений ГИБДД и их подчиненность

В составе ГИБДД дорожно-патрульная служба формируется преимущественно в виде строевых подразделений. Термин «строевые» означает, что их структура, внутренняя организация и система подчиненности близки к воинским формированиям.

В зависимости от количества зарегистрированных ТС, протяженности автомобильных дорог, уровня ДТА, а также наличия необходимых ресурсов в пределах административно-территориальных единиц могут комплектоваться следующие подразделения ДПС:

* группа;
* отдельный взвод;
* моторизованный взвод (группа);
* отдельная рота;
* отдельный батальон;
* полк.

В соответствие с типовыми штатами строевых подразделений ДПС ГИБДД делится на группы, отдельные и моторизованные взводы численностью до 50 человек обычно создаются при районных и городских ОВД. Более крупные подразделения – отдельные роты, батальоны и полки ДПС – формируются в республиках, краях, областях и крупных городах. Как правило, они подчиняются по вертикали – вышестоящим органам ГИБДД (отделам или управлениям ГИБДД УВД МВД субъектов Российской Федерации) и по горизонтали – территориальным органам внутренних дел (УВД ГОРОВД, Рисунок 1).



Рисунок – Схема подчиненности ДПС

### Бизнес-модель ГИБДД

Разработка моделей в стандарте IDEF0 позволяет наглядно и эффективно отобразить весь механизм деятельности предприятия в нужном разрезе. Разработанная бизнес-модель демонстрируется на диаграммах:

* контекстная диаграмма ГИБДД (Рисунок 2);
* диаграмма подсистем ГИБДД (Рисунок 3);

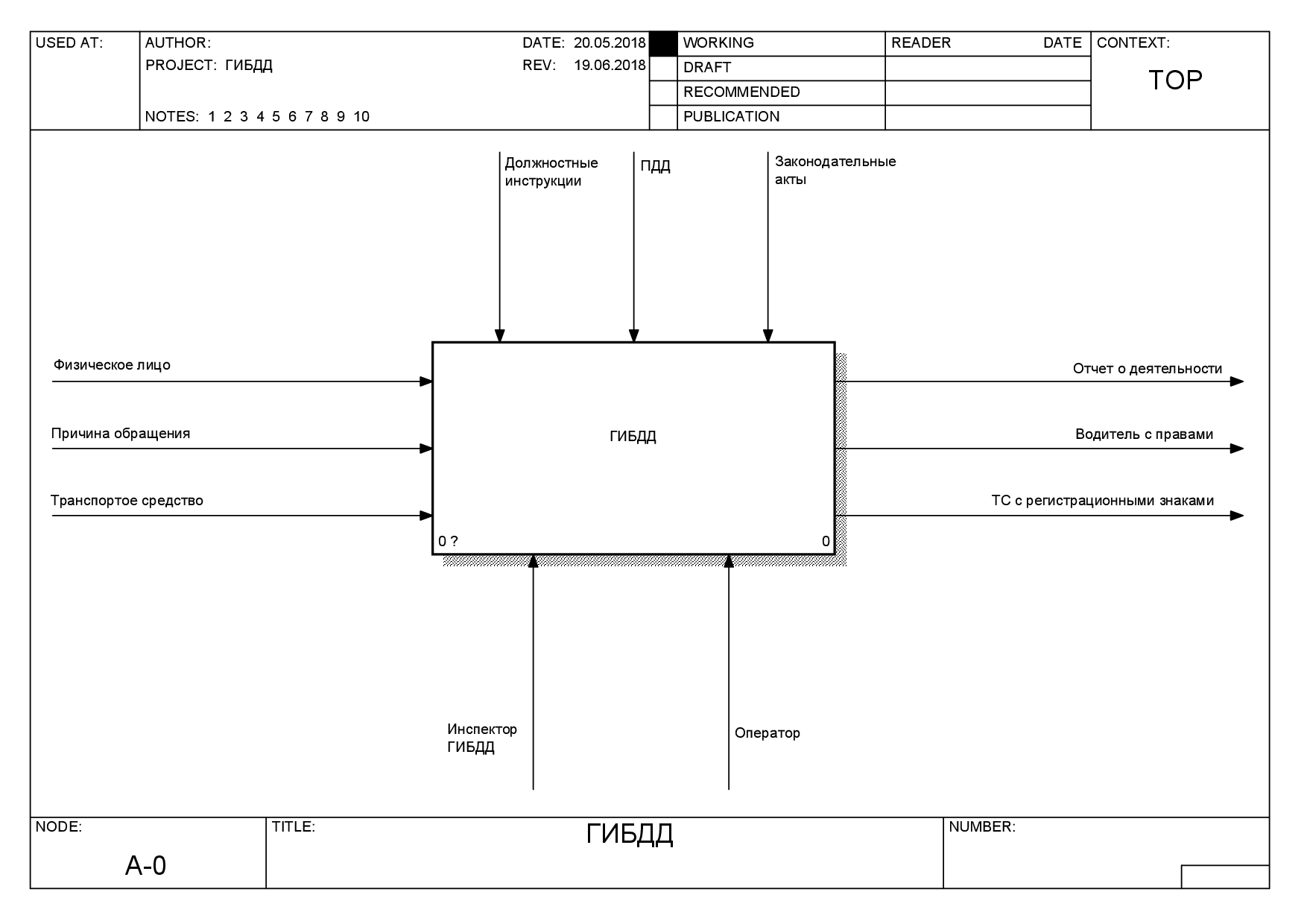


Рисунок – Контекстная диаграмма процессов ГИБДД

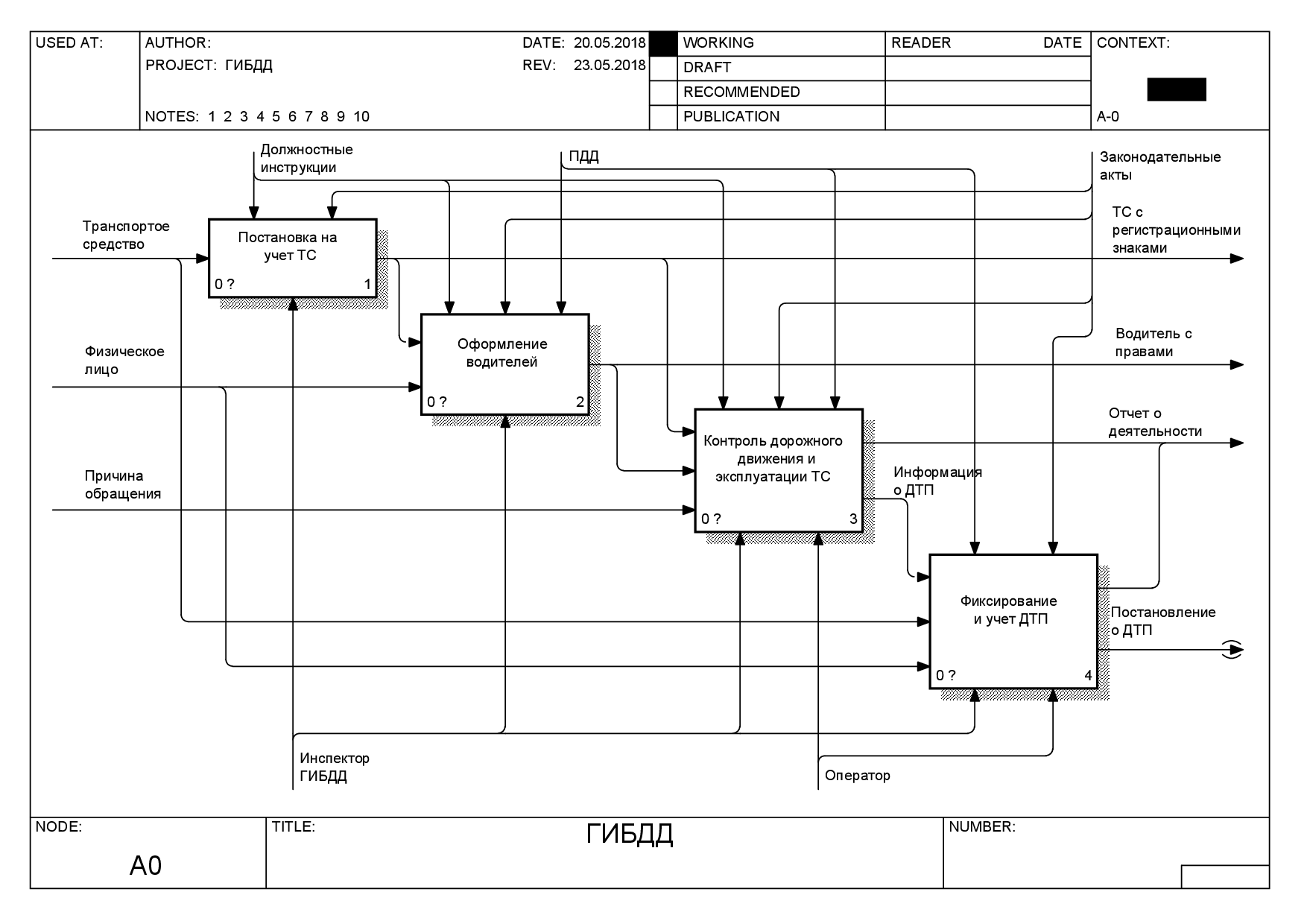


Рисунок – Диаграмма подсистем ГИБДД

Модели были спроектированы при помощи средств визуального моделирования бизнес-процессов BPwin.

### Потоки данных ГИБДД

Диаграммы потоков данных (DFD) представляют собой иерархию функциональных процессов, связанных потоками данных. Цель такого представления – продемонстрировать, как каждый процесс преобразует свои входные данные в выходные, а также выявить отношения между этими процессами.

Основными компонентами, которыми проводится описание диаграмм потоков данных, являются: внешние сущности, системы и подсистемы, процессы, накопители данных, потоки данных.

Контекстная диаграмма (Рисунок 4), и диаграмма подсистем ПС (Рисунок 5) помогают проектировщику лучше выбрать процесс для автоматизации.

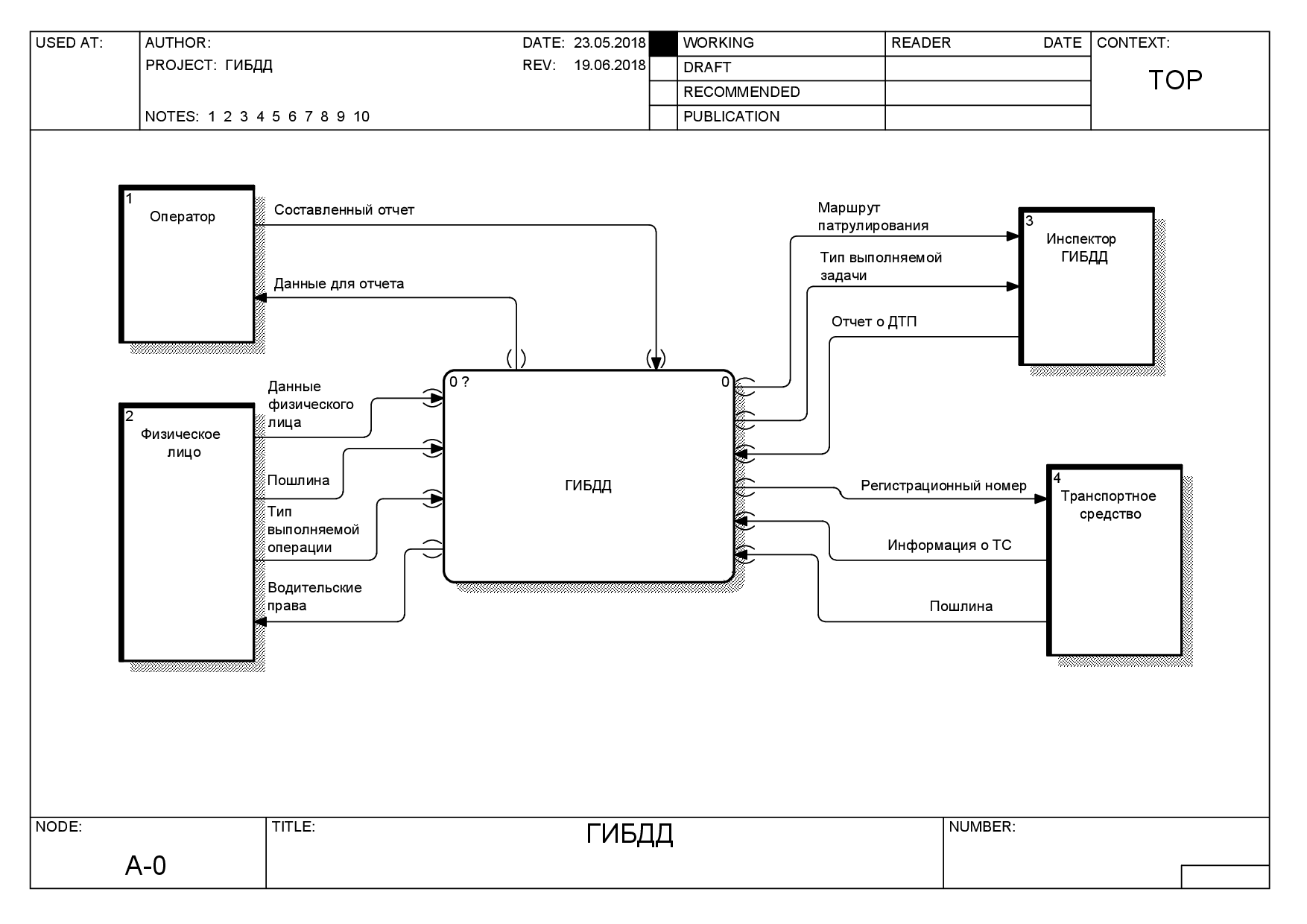


Рисунок – Контекстная диаграмма потоков ГИБДД

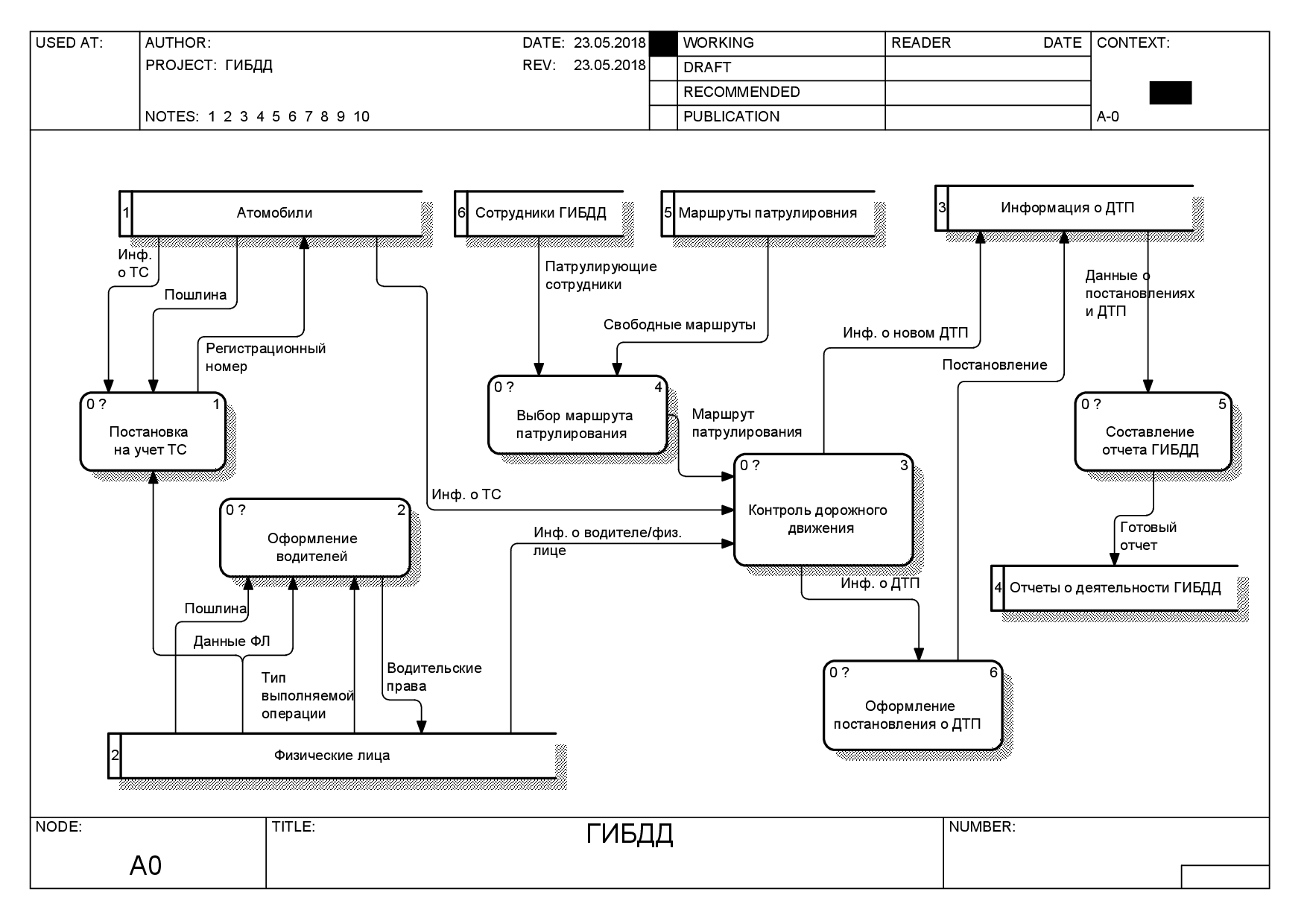


Рисунок – Диаграмма потоков информации в ГИБДД

## Обоснование решаемой задачи

В результате проведенного анализа было выявлен неоптимальный бизнес-процесс на уровне модели «Фиксирование и учет ДТП». Формирование постановления о ДТП происходит вручную, что занимает много времени. Данный процесс поддается автоматизации.

Целью разработки программного средства выбрана автоматизация оформления постановления о ДТП.

Необходимо хранить все сведения о ДТП (место и время происшествия, пункты нарушения), сведения об участниках (ФИО, паспортные данные, год и место рождения, статус в ДТП), сведения об участвующих ТС (ПТС, VIN, цвет, марка, тип, гос. номер, владелец), также следует хранить информацию о составителе документа (ФИО, должность, звание).

Все эти сведения будут фигурировать в итоговом документе – постановлении о ДТП.

# Проектная часть

## Назначение базы данных

### Описание выходной информации

Результатом разработки программного средства должно быть постановление о ДТП. Постановление должно представлять собой документ, отражающий следующую информацию:

* дата и место составления документа;
* должность, фамилия и инициалы инспектора ГИБДД;
* данные о владельце транспортного средства;
* данные о свидетелях и потерпевших, в частности их ФИО и адреса;
* показания свидетелей;
* указание нормативно-правового акта, который был нарушен, место и время совершения правонарушения, а также подробное описание ситуации и роли в ней участников нарушения.
* указание статьи, которая предусматривает ответственность за правонарушение;
* объяснение лица, в отношении которого возбуждено дело;
* иные сведения, которые могут пригодиться для разрешения дела;
* форма административного наказания.

### Описание входной информации

Для осуществления программным средством возложенной на нее задачи, необходимо ввести следующие данные:

Информация о физических лицах, фигурирующих в участниках ДТП и владельцах авто (ФИО, № паспорта, Адрес проживания, Дата рождения).

Информация об актуальных водительских правах (№ паспорта водителя, № водительских прав).

Информация об автомобилях (ПТС, Модель, Цвет, VIN, № кузова, № двигателя, Год выпуска). Если автомобиль имеет регистрационный знак, то требуется хранить и его.

Информацию о сотрудниках ГИБДД (ФИО, Должность, Звание).

Информацию о типах возможных правонарушений (Название нарушения, Название, Размер штрафа, КоАП).

Описанной выше информации будет достаточно для составления постановления и формирования отчетов о деятельности ГИБДД.

## Проектирование базы данных

### Концептуальное проектирование

На этом этапе создаются подробные модели пользовательских представлений данных предметной области. Затем они интегрируются в концептуальную модель, которая фиксирует все элементы корпоративных данных, подлежащих загрузке в базу данных. Эту модель называют концептуальной схемой базы данных.

Средством моделирования предметной области на этапе концептуального проектирования является модель «сущность-связь». Часто ее называют ER-моделью. В ней моделирование структуры данных предметной области базируется на использовании графических средств – ER-диаграмм. В наглядном виде они представляют связи между сущностями.

Основными понятиями ER-диаграммы являются сущность, атрибут, связь.

Сущность представляет собой объект, информация о котором хранится в базе данных. Сущность имеет экземпляры, отличающиеся друг от друга значениями атрибутов и допускающие однозначную идентификацию.

Атрибут – это свойство сущности. Атрибут, который уникальным образом идентифицирует экземпляры сущности, называется ключом. Может быть составной ключ, представляющий комбинацию нескольких атрибутов.

Связь представляет взаимодействие между сущностями. Она характеризуется мощностью, которая показывает, сколько сущностей участвует в связи.

Создадим ER-модель предметной области ГИБДД (Рисунок 1). Каждая из сущностей приведенной ER-модели может быть описана своим набором атрибутов. Первичные ключи для сущностей выделим жирным шрифтом.



Рисунок – ER-диаграмма предметной области

ER-модель в совокупности с наборами атрибутов сущностей может служить примером концептуальной модели предметной области или концептуальной схемы базы данных.

Рассмотрим проектирование базы данных ГИБДД. В ней могут быть определены следующие сущности:

* цвет – перечисление всех возможных цветов автомобилей;
* модель – информация о модели и категории автомобиля;
* машина – описание конкретного автомобиля;
* физическое лицо – информацию о конкретном человеке;
* водитель – граждане с водительским удостоверением;
* свидетельство о регистрации ТС – ассоциирует авт. с гос. номером;
* сотрудник – все сотрудники ГИБДД;
* нарушение – возможные типы нарушений ПДД;
* тип участника – определение статуса участника при ДТП;
* дтп – информация о постановлении при ДТП;
* участники автомобили – ассоциирует автомобиль с постановлением;
* участники пешеходы – ассоциирует физ. лицо с постановлением;
* штрафы – ассоциирует нарушение с постановлением;

Набор атрибутов для каждой сущности проще всего представить в табличном виде (Таблица 1). Название таблиц выделены нижним подчеркивание, **первичные ключи** – жирным, *внешние ключи* – курсивом.

Таблица – Наборы атрибутов сущностей предметной области

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | Цвет | | **№ цвета** | | Название | | |  | | --- | | Модель | | **№ модели** | | Категория | | Наименование | | |  | | --- | | Машина | | **ПТС** | | *№ модели* | | *№ цвета* | | VIN | | № кузова | | № двигателя | | Год выпуска | | |  | | --- | | Физическое лицо | | **Паспорт** | | ФИО | | Адр. проживания | | Дата рождения | |
| |  | | --- | | Водитель | | *Паспорт* | | Водительское удостоверение | | | |  | | --- | | Свидетельство о регистрации ТС | | **Госномер** | | *Паспорт* | | *ПТС* | | |
| |  | | --- | | Нарушение | | **№ нарушения** | | Название | | Размер штрафа | | КоАП | | |  | | --- | | Тип участника | | **№ типа** | | Статус | | |  | | --- | | Штрафы | | *№нарушения* | | *Постановление* | | Надбавка | | |  | | --- | | Сотрудник | | **№ сотрудника** | | ФИО | | Должность | | Звание | |
| |  | | --- | | ДТП | | **Постановление** | | *№ сотрудника* | | Место происшествия | | Описание | | Дата | | | |  | | --- | | Участники автомобили | | *Паспорт* | | *Постановление* | | *Госномер* | | *№ типа* | | |
| |  | | --- | | Участники пешеходы | | *Паспорт* | | *Постановление* | | *№ типа* | | |  | |

### Логическое проектирование

Концептуальные модели позволяют более точно представить предметную область, чем реляционные и другие более ранние модели. Но в настоящее время существует немного систем управления базами данных, поддерживающих эти модели. На практике наиболее распространены системы, реализующие реляционную модель (Рисунок 2). Поэтому необходим метод перевода концептуальной модели в реляционную. Такой метод основывается на формировании набора предварительных таблиц из ER-диаграмм.

Для каждой сущности создается таблица. Причем каждому атрибуту сущности соответствует столбец таблицы.

Правила генерации таблиц из ER-диаграмм опираются на два основных фактора – тип связи и класс принадлежности сущности.

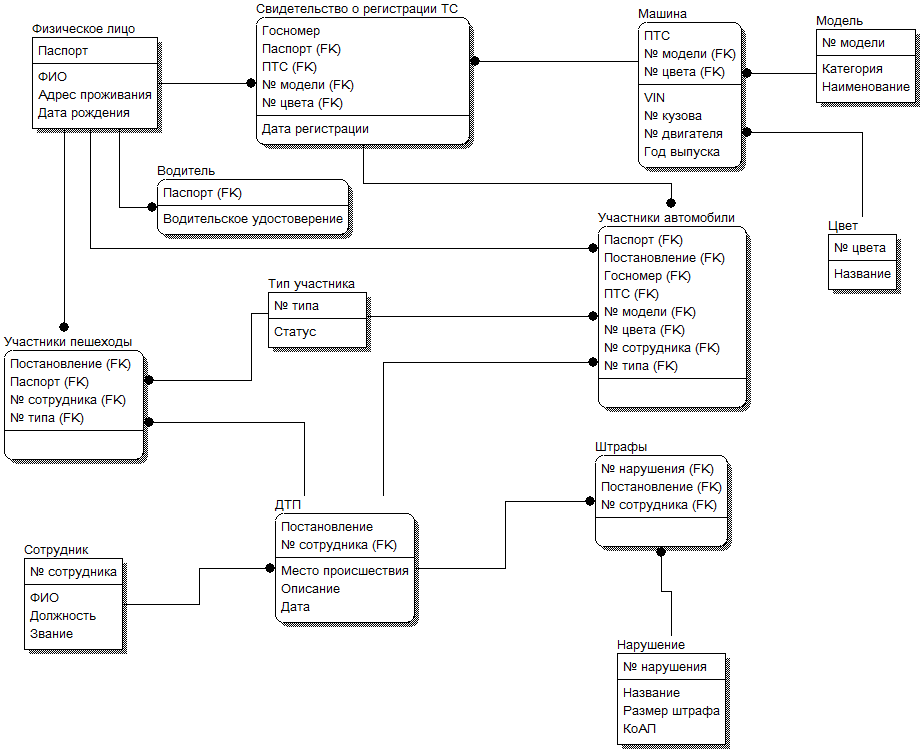


Рисунок – Логическая модель предметной области

Т. к. в ER-диаграмме отношения многие ко многим уже были развернуты в отдельные сущности – ассоциации, то логическая модель была построена на основе ER-диаграммы практически без изменений.

### Физическое проектирование

Цель этапа физического проектирования – описание конкретной реализации базы данных, размещаемой во внешней памяти компьютера. Это описание структуры хранения данных и эффективных методов доступа к данным базы. При логическом проектировании отвечают на вопрос – что надо сделать, а при физическом – выбирается способ, как это сделать.

Для проектирования таблиц базы данных будем использовать компактную встраиваемую СУБД SQLite (Рисунок 3).

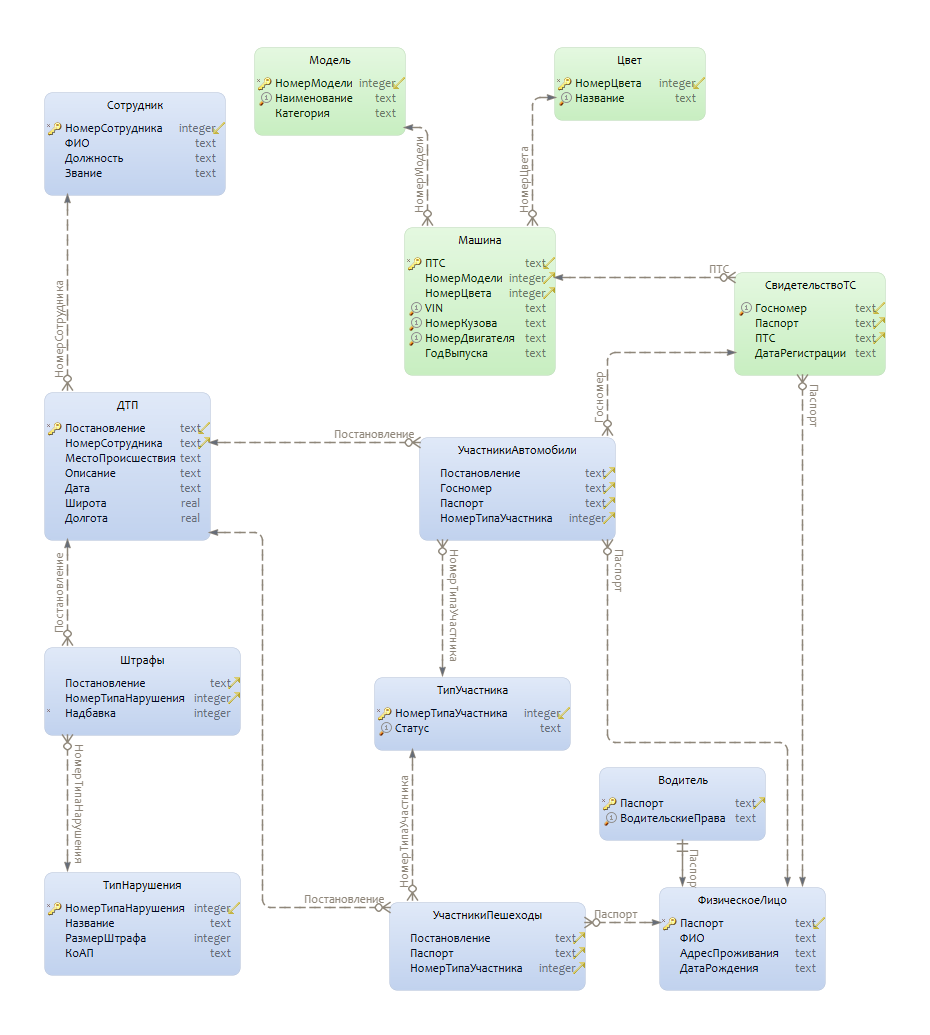


Рисунок – Физическая модель предметной области в SQLite

*Код генерации DDL:*

--

-- Использованная кодировка текста: UTF-8

--

**PRAGMA** foreign\_keys **=** **off;**

**BEGIN** **TRANSACTION;**

-- Таблица: Водитель

**CREATE** **TABLE** В**о**дите**л**ь **(**Пас**по**рт **VARCHAR** **PRIMARY** **KEY** **REFERENCES** Физичес**ко**еЛиц**о** **(**Пас**по**рт**)** **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE** **NOT** **NULL,** В**о**дите**л**ьс**к**иеПрава **VARCHAR** **UNIQUE);**

-- Таблица: ДТП

**CREATE** **TABLE** ДТП **(**П**о**ста**но**в**л**е**н**ие **VARCHAR** **PRIMARY** **KEY** **NOT** **NULL,** Н**ом**ер**Со**труд**н**и**к**а **VARCHAR** **REFERENCES** **Со**труд**н**и**к** **(**Н**ом**ер**Со**труд**н**и**к**а**)** **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE,** Мест**о**Пр**о**исшествия **VARCHAR,** О**п**иса**н**ие **VARCHAR,** Дата **VARCHAR** **DEFAULT** **(**strftime**(**'%d.%m.%Y'**,** 'now'**)),** **Ш**ир**о**та **REAL,** Д**ол**г**о**та **REAL);**

-- Таблица: Машина

**CREATE** **TABLE** Маши**н**а **(**ПТ**С** **VARCHAR** **PRIMARY** **KEY** **NOT** **NULL,** Н**ом**ерМ**о**де**л**и **INTEGER** **REFERENCES** М**о**де**л**ь **(**Н**ом**ерМ**о**де**л**и**)** **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE,** Н**ом**ер**Ц**вета **INTEGER** **REFERENCES** **Ц**вет **(**Н**ом**ер**Ц**вета**)** **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE,** VIN **VARCHAR** **UNIQUE,** Н**ом**ерКуз**о**ва **VARCHAR** **UNIQUE,** Н**ом**ерДвигате**л**я **VARCHAR** **UNIQUE,** Г**о**дВы**п**ус**к**а **VARCHAR);**

-- Таблица: Модель

**CREATE** **TABLE** М**о**де**л**ь **(**Н**ом**ерМ**о**де**л**и **INTEGER** **NOT** **NULL** **PRIMARY** **KEY,** Наи**м**е**но**ва**н**ие **VARCHAR** **UNIQUE,** Катег**о**рия **VARCHAR);**

-- Таблица: СвидетельствоТС

**CREATE** **TABLE** **С**видете**л**ьств**о**Т**С** **(**Г**о**с**ном**ер **VARCHAR** **PRIMARY** **KEY,** Пас**по**рт **VARCHAR** **REFERENCES** Физичес**ко**еЛиц**о** **(**Пас**по**рт**)** **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE,** ПТ**С** **VARCHAR** **REFERENCES** Маши**н**а **(**ПТ**С)** **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE,** ДатаРегистрации **VARCHAR);**

-- Таблица: Сотрудник

**CREATE** **TABLE** **Со**труд**н**и**к** **(**Н**ом**ер**Со**труд**н**и**к**а **INTEGER** **PRIMARY** **KEY** **NOT** **NULL,** ФИО **VARCHAR,** Д**ол**ж**но**сть **VARCHAR,** Зва**н**ие **VARCHAR);**

-- Таблица: ТипНарушения

**CREATE** **TABLE** Ти**п**Наруше**н**ия **(**Н**ом**ерТи**п**аНаруше**н**ия **INTEGER** **NOT** **NULL** **PRIMARY** **KEY,** Назва**н**ие **VARCHAR,** Раз**м**ер**Ш**трафа **INTEGER,** К**о**АП **VARCHAR);**

-- Таблица: ТипУчастника

**CREATE** **TABLE** Ти**п**Участ**н**и**к**а **(**Н**ом**ерТи**п**аУчаст**н**и**к**а **INTEGER** **PRIMARY** **KEY** **NOT** **NULL,** **С**татус **VARCHAR** **UNIQUE);**

-- Таблица: УчастникиАвтомобили

**CREATE** **TABLE** Участ**н**и**к**иАвт**омо**би**л**и **(**П**о**ста**но**в**л**е**н**ие **VARCHAR** **REFERENCES** ДТП **(**П**о**ста**но**в**л**е**н**ие**)** **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE,** Г**о**с**ном**ер **VARCHAR** **REFERENCES** **С**видете**л**ьств**о**Т**С** **(**Г**о**с**ном**ер**)** **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE,** Пас**по**рт **VARCHAR** **REFERENCES** Физичес**ко**еЛиц**о** **(**Пас**по**рт**)** **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE,** Н**ом**ерТи**п**аУчаст**н**и**к**а **INTEGER** **REFERENCES** Ти**п**Участ**н**и**к**а **(**Н**ом**ерТи**п**аУчаст**н**и**к**а**)** **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE);**

-- Таблица: УчастникиПешеходы

**CREATE** **TABLE** Участ**н**и**к**иПешех**о**ды **(**П**о**ста**но**в**л**е**н**ие **VARCHAR** **REFERENCES** ДТП **(**П**о**ста**но**в**л**е**н**ие**)** **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE,** Пас**по**рт **VARCHAR** **REFERENCES** Физичес**ко**еЛиц**о** **(**Пас**по**рт**)** **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE,** Н**ом**ерТи**п**аУчаст**н**и**к**а **INTEGER** **REFERENCES** Ти**п**Участ**н**и**к**а **(**Н**ом**ерТи**п**аУчаст**н**и**к**а**)** **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE);**

-- Таблица: ФизическоеЛицо

**CREATE** **TABLE** Физичес**ко**еЛиц**о** **(**Пас**по**рт **VARCHAR** **PRIMARY** **KEY** **NOT** **NULL,** ФИО **VARCHAR,** АдресПр**о**жива**н**ия **VARCHAR,** ДатаР**о**жде**н**ия **VARCHAR);**

-- Таблица: Цвет

**CREATE** **TABLE** **Ц**вет **(**Н**ом**ер**Ц**вета **INTEGER** **NOT** **NULL** **PRIMARY** **KEY,** Назва**н**ие **VARCHAR** **UNIQUE);**

-- Таблица: Штрафы

**CREATE** **TABLE** **Ш**трафы **(**П**о**ста**но**в**л**е**н**ие **VARCHAR** **REFERENCES** ДТП **(**П**о**ста**но**в**л**е**н**ие**)** **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE,** Н**ом**ерТи**п**аНаруше**н**ия **INTEGER** **REFERENCES** Ти**п**Наруше**н**ия **(**Н**ом**ерТи**п**аНаруше**н**ия**)** **ON** **DELETE** **CASCADE** **ON** **UPDATE** **CASCADE,** Надбав**к**а **INTEGER** **NOT** **NULL** **DEFAULT** **(**0**));**

**COMMIT** **TRANSACTION;**

**PRAGMA** foreign\_keys **=** **on;**

# Клиентское приложение

Клиентское приложение для БД ГИБДД реализовано на Lazarus версии 1.8.4 с использованием сервисов Google Maps и дополнительных компонентов: fpCEF (для отображения карт) и LazReport (для визуализации формы постановления). Сама БД была реализована в СУБД SQLite при помощи SQLiteStudio версии 3.1.1. Для решения задач управления постановлениями о ДТП, создано несколько форм с удобным интерфейсом, позволяющие решать типовые задачи. Он состоит из таблиц, для вывода данных из БД, элементов управления данными и навигации по таблице. Каждая форма имеет собственный функциональный набор.

Навигация по существующим формам осуществляется через меню основного окна программы (Рисунок 9).

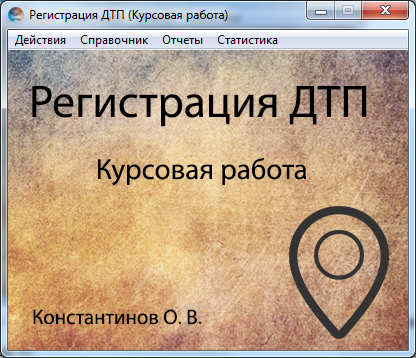


Рисунок – Главное окно программы

В качестве выходной информации выступает постановление о ДТП (Рисунок 10), данные в котором формируются в СУБД при помощи представления «ОформленныеПостановления», а отображаются в форме клиентского приложения при помощи компонента LazReport.

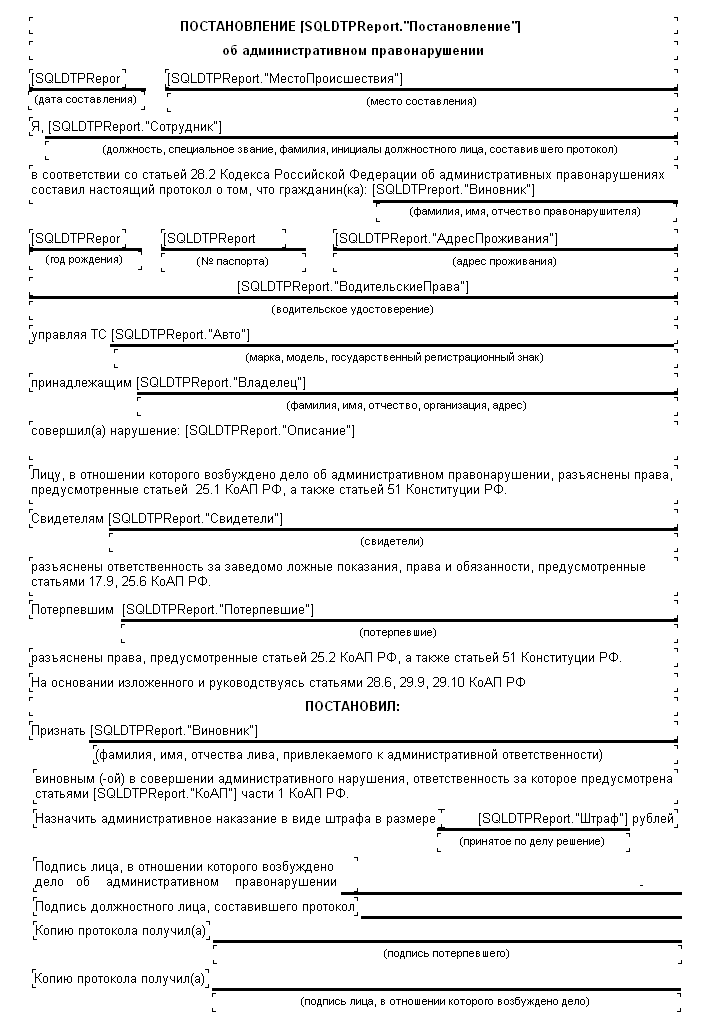


Рисунок – Форма постановления о ДТП

*Код представления «АвтоПоГосНомеру»:*

SELECT стс.Госномер,

мд.Наименование || ', цвет: ' || ц.Название || ', госномер: ' || стс.Госномер || ', год выпуска: ' || м.ГодВыпуска Авто,

фл.ФИО || ', дата рождения: ' || фл.ДатаРождения || ', паспорт: ' || фл.Паспорт Владелец

FROM СвидетельствоТС стс

INNER JOIN

Машина м ON м.ПТС = стс.ПТС

INNER JOIN

Модель мд ON мд.НомерМодели = м.НомерМодели

INNER JOIN

Цвет ц ON ц.НомерЦвета = м.НомерЦвета

INNER JOIN

ФизическоеЛицо фл ON фл.Паспорт = стс.Паспорт

*Код представления «УчастникиПоПостановлению»:*

SELECT уч.Постановление,

фл.ФИО,

уч.Авто,

уч.Владелец,

фл.Паспорт,

фл.ДатаРождения,

фл.АдресПроживания,

в.ВодительскиеПрава,

ту.Статус

FROM (

SELECT Постановление,

Паспорт,

Авто,

Владелец,

НомерТипаУчастника

FROM УчастникиАвтомобили уа

INNER JOIN

АвтоПоГосНомеру оапгн ON уа.Госномер = оапгн.Госномер

UNION ALL

SELECT Постановление,

Паспорт,

NULL,

NULL,

НомерТипаУчастника

FROM УчастникиПешеходы

)

уч

INNER JOIN

ФизическоеЛицо фл ON фл.Паспорт = уч.Паспорт

INNER JOIN

ТипУчастника ту ON ту.НомерТипаУчастника = уч.НомерТипаУчастника

LEFT JOIN

Водитель в ON в.Паспорт = фл.Паспорт

*Код представления «ШтрафПоПостановлению»:*

SELECT д.Постановление,

SUM(тн.РазмерШтрафа + ш.Надбавка) Штраф,

GROUP\_CONCAT(тн.КоАП, ', ') КоАП

FROM ДТП д

LEFT JOIN

Штрафы ш ON ш.Постановление = д.Постановление

LEFT JOIN

ТипНарушения тн ON тн.НомерТипаНарушения = ш.НомерТипаНарушения

GROUP BY д.Постановление

Все вышеописанные представления служат для формирования составного представления по совершенным ДТП.

*Код представления «ОформленныеПостановления»:*

**SELECT** с**.**ФИО **||** ', ' **||** с**.**Д**ол**ж**но**сть **||** ', ' **||** с**.**Зва**н**ие **Со**труд**н**и**к,**

д**.**П**о**ста**но**в**л**е**н**ие**,**

д**.**Мест**о**Пр**о**исшествия**,**

д**.**О**п**иса**н**ие**,**

д**.**Дата**,**

у**пп.**ФИО Ви**но**в**н**и**к,**

у**пп.**Пас**по**рт**,**

у**пп.**ДатаР**о**жде**н**ия**,**

у**пп.**АдресПр**о**жива**н**ия**,**

у**пп.**Авт**о,**

у**пп.**В**л**аде**л**ец**,**

у**пп.**В**о**дите**л**ьс**к**иеПрава**,**

ш**пп.Ш**траф**,**

ш**пп.**К**о**АП**,**

**(**

**SELECT** GROUP\_CONCAT**(**ФИО **||** ', п.: ' **||** Пас**по**рт**,** '; '**)**

**FROM** Участ**н**и**к**иП**о**П**о**ста**но**в**л**е**н**ию

**WHERE** **С**татус **=** 'свидетель' **AND**

П**о**ста**но**в**л**е**н**ие **=** д**.**П**о**ста**но**в**л**е**н**ие

**GROUP** **BY** **С**татус

**LIMIT** 2

**)**

**С**видете**л**и**,**

**(**

**SELECT** GROUP\_CONCAT**(**ФИО **||** ', п.: ' **||** Пас**по**рт**,** '; '**)**

**FROM** Участ**н**и**к**иП**о**П**о**ста**но**в**л**е**н**ию

**WHERE** **С**татус **=** 'потерпевший' **AND**

П**о**ста**но**в**л**е**н**ие **=** д**.**П**о**ста**но**в**л**е**н**ие

**GROUP** **BY** **С**татус

**LIMIT** 2

**)**

П**о**тер**п**евшие

**FROM** ДТП д

**INNER** **JOIN**

**Со**труд**н**и**к** с **ON** с**.**Н**ом**ер**Со**труд**н**и**к**а **=** д**.**Н**ом**ер**Со**труд**н**и**к**а

**INNER** **JOIN**

**(**

**SELECT** **\***

**FROM** Участ**н**и**к**иП**о**П**о**ста**но**в**л**е**н**ию

**WHERE** **С**татус **=** 'виновник'

**)**

у**пп** **ON** у**пп.**П**о**ста**но**в**л**е**н**ие **=** д**.**П**о**ста**но**в**л**е**н**ие

**INNER** **JOIN**

**Ш**трафП**о**П**о**ста**но**в**л**е**н**ию ш**пп** **ON** ш**пп.**П**о**ста**но**в**л**е**н**ие **=** д**.**П**о**ста**но**в**л**е**н**ие

Результатом исполнения представления «ОформленныеПостановления» будет список ДТП со следующими полями:

* Сотрудник.
* Постановление.
* МестоПроисшествия.
* Описание.
* Дата.
* ФИО Виновник.
* Паспорт.
* ДатаРождения.
* АдресПроживания.
* Авто.
* Владелец.
* ВодительскиеПрава.
* Штраф.
* КоАП.

# Заключение

В результате проведенных работ по курсовому проекту, была спроектирована и реализована информационная система и база дынных. Внедрение данной информационной системе позволит более оперативно составлять постановления сотрудниками ГИБДД снизить случайные ошибки связанных с ручным заполнением форм протоколов.

Для реализации курсового проекта использовались средства проектирования BPWin и ERWin в качестве среды построения моделей, IDE Lazarus для реализации оболочки программного средства, SQLiteStudio для создания БД в СУБД SQLite.

В результате проведенной работы была достигнутая поставленная цель: разработка проекта программного средства для ГИБДД.

# Список литературы

**Абдулаев В.И.** Программная инженерия: учебное пособие [Книга]. - Йошкар-Ола : ПГТУ, 2016. - стр. 168.

**Александров Д.В** Инструментальные средства информационного менеджмента. CASE-технологии и распределенные информационные системы [Книга]. - Москва : Финансы и статистика, 2011. - стр. 224.

**Антамошкин О.А.** Программная инженерия. Теория и практика: учебник [Книга]. - Красноярск : СФУ, 2012. - стр. 247.

**Батоврин В.К.** Системная и программная инженерия. Словарь-справочник: учебное пособие для вузов [Книга]. - Москва : ДМК Пресс, 2010. - стр. 280.

**Ехлаков Ю.П.** Введение в программную инженерию [Книга]. - Москва : ТУСУР, 2012. - стр. 14.

**Маклаков С.В.** Моделирование бизнес-процессов с BPwin 4.0 [Книга]. - Москва : Диалог-МИФИ, 2002. - стр. 224.

**Тельнов Ю.Ф.** Реинжиниринг бизнес-процессов [Книга]. - Москва : Финансы и статистика, 2005. - стр. 320.

**Хаммер М.** Быстрее, дешевле: Девять методов реинжиниринга бизнес-процессов [Книга]. - Москва : Альпина Паблишер, 2016. - стр. 352.