



ИНТЕЛТРАНС
Г Р У П П А К О М П А Н И Й

IT & TRANSPORT

ИТ & ТРАНСПОРТ

Сетевое издание

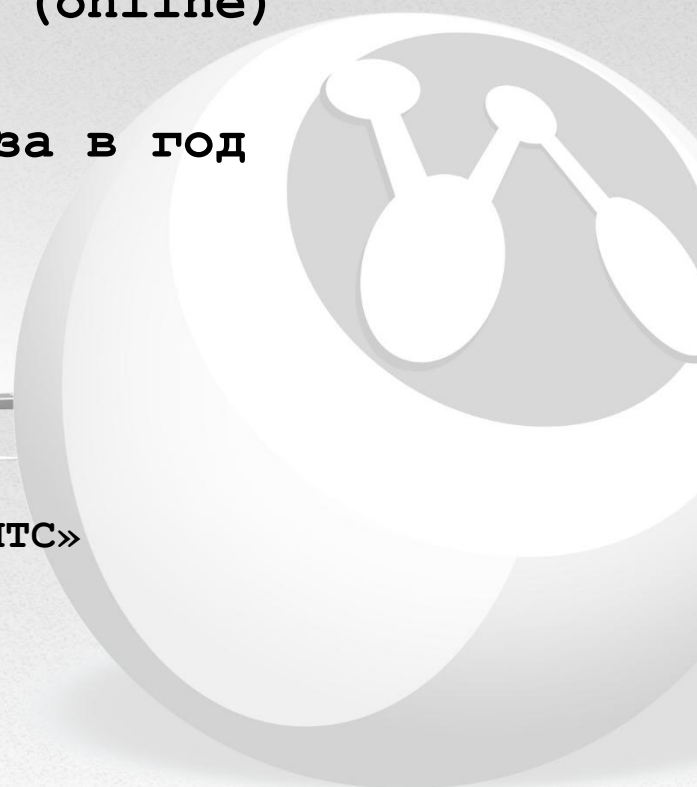
№ 4 (28) – 2024

ISSN 2949–5288 (online)

Выходит: 4 раза в год

Самара
ООО «НПЦ «ИТС»

2024





ИНТЕЛТРАНС

Г Р У П П А К О М П А Н И Й

И Т & Т Р А Н С П О Р Т

И Т & Т Р А Н С П О Р Т

Направления :

**1.2.2. МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ, ЧИСЛЕННЫЕ
МЕТОДЫ И КОМПЛЕКСЫ ПРОГРАММ (технические науки)**

**2.3.1. СИСТЕМНЫЙ АНАЛИЗ, УПРАВЛЕНИЕ И ОБРАБОТКА
ИНФОРМАЦИИ (технические науки)**

**2.3.3. АВТОМАТИЗАЦИЯ И УПРАВЛЕНИЕ ТЕХНОЛОГИЧЕСКИМИ
ПРОЦЕССАМИ И ПРОИЗВОДСТВАМИ (технические науки)**

**2.3.5. МАТЕМАТИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ
ВЫЧИСЛИТЕЛЬНЫХ МАШИН, КОМПЛЕКСОВ И КОМПЬЮТЕРНЫХ
СЕТЕЙ (технические науки)**

**Главный редактор Михеева Т. И.
доктор технических наук, профессор,
заслуженный деятель науки и образования РФ**

**Самара
ООО «НПЦ «ИТС»
2024**

IT & Transport / ИТ & Транспорт : под ред. профессора Т. И. Михеевой. – Самара, ООО «НПЦ «ИТС», 2024, №4 (28). – 73 с. ил.

ISSN: 2949-5288

В сборнике представлены научные статьи, содержащие результаты исследований в следующих предметных областях: системный анализ, управление и обработка информации, интеллектуальные транспортные системы, геоинформационные системы, информационные технологии, базы данных и знаний, системы искусственного интеллекта, цифровая обработка изображений, управление транспортными процессами, развитие транспортной инфраструктуры, автоматизация и управление технологическими процессами и производствами, математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей, математическое и компьютерное моделирование.

Предназначен для научно-технических работников, инженеров, аспирантов и студентов, обучающихся по специальностям, связанным с информационными технологиями, транспортными системами и процессами.

Авторская позиция и стилистические особенности публикаций полностью сохранены.

Главный редактор – Т. И. Михеева доктор технических наук, профессор,
заслуженный деятель науки и образования РФ

Редакционная коллегия:

Бурдин А.В.	доктор технических наук, доцент
Гераськин М.И.	доктор экономических наук, профессор
Жанказиев С.В.	доктор технических наук, профессор
Заболотнов Ю.М.	доктор технических наук, профессор
Зырянов В.В.	доктор технических наук, профессор
Иващенко А.В.	доктор технических наук, профессор
Минниханов Р.Н.	доктор технических наук, профессор
Прохоров С.А.	доктор технических наук, профессор
Филиппова А.С.	доктор технических наук, профессор
Хайтбаев В.А.	доктор экономических наук, профессор
Зеленко Л.С.	кандидат технических наук, доцент
Золотовицкий А.В.	кандидат технических наук, доцент
Михеев С.В.	кандидат технических наук, доцент
Сапрыкин О.Н.	кандидат технических наук, доцент
Сапрыкина О.В.	кандидат технических наук, доцент
Федосеев А.А.	кандидат технических наук, доцент
Шопин А.Г.	кандидат технических наук

Редакция: Чекина Е.В.

ISBN 978-5-6049535-7-0©

Сетевое издание «ИТ & Транспорт» зарегистрировано Федеральной службой по надзору в сфере связи, информационных технологий и массовых коммуникаций; регистрационный номер: серия ЭЛ № ФС 77–85276 от 30.05.2023

e-mail: itsspc@yandex.ru

© Адрес страницы сайта: <http://ittransport.ru>

© ООО «НПЦ «ИТС», 2024

УДК 004.02

Михеева Т.И., Клепиков Н.М.

КЛАССИФИКАЦИЯ БЕСПИЛОТНЫХ ЛЕТАТЕЛЬНЫХ АППАРАТОВ

*Самарский университет имени академика С.П. Королёва
ИнтелТранс*

В статье рассматривается классификация беспилотных летательных аппаратов. Отмечаются разновидности, по которым для решения задачи распознавания объектов желательно использовать беспилотные летательные аппараты.

Ключевые слова: беспилотный летательный аппарат, классификация, обнаружение объектов, ортоизображение, электронная карта.

Введение

Пилотируемые летательные аппараты существуют уже давно, и отрасль разработала хорошие системы для их классификации. FAA классифицирует пилотируемые летательные аппараты по категориям, классам и типам. Основные категории и классы гражданских пилотируемых летательных аппаратов:

Самолеты:

- Одномоторные наземные.
- Многомоторные наземные.
- Одномоторные морские.
- Многомоторные морские.

Вертолет:

- Вертолет.
- Жироплан.
- Конвертоплан.

Легче воздуха или аэростаты:

- Дирижабль (например, дирижабль).
- Воздушный шар (например, воздушный шар).

Планер.

Другие категории летательных аппаратов включают:

С подъемной силой (включая беспилотные летательные аппараты или БПЛА).

С парашютами:

- Наземные операции.

- Морские операции.

Самолеты со смещением веса (например, дельтапланы):

- Наземные операции.
- Морские операции.

Ракеты

Когда речь идет о БПЛА, в отрасли нет единого стандарта классификации, поскольку использование и технологии БПЛА быстро развиваются. У оборонных ведомств есть свой собственный стандарт, а у гражданских лиц есть свои постоянно развивающиеся свободные категории для БПЛА. Люди классифицируют их по размеру, дальности и выносливости и используют систему уровней, которая применяется военными. Веб-сайт Управления интеграции национальной авиационной разведки США предоставляет хороший обзор глобальных категорий классификации БПЛА. Для классификации по размеру можно выделить следующие подклассы:

- Очень малые БПЛА.
- Микро- или нано-БПЛА.
- Малые БПЛА.
- Мини-БПЛА.
- Средние БПЛА.
- Крупные БПЛА.

БПЛА также можно классифицировать по дальности полета и продолжительности полета, используя следующие подклассы, разработанные военными США:

- БПЛА ближнего действия
- БПЛА среднего действия
- БПЛА дальнего действия

По данным Министерства обороны США, БПЛА подразделяются на пять категорий, как показано в Таблице 1:

Таблица 1. Классификация БПЛА по данным Министерства обороны США

Категория	Размер	Максимальный взлетный вес, фунты	Нормальная рабочая высота, футы	Скорость полета, узлы
Группа 1	Маленький	0-20	<1200	<100
Группа 2	Средний	21-55	<3500	<250
Группа 3	Большой	<1320	<18000	<250
Группа 4	Большой	>1320	<18000	Любая воздушная скорость
Группа 5	Самый большой	>1320	>18000	Любая воздушная скорость

Классификация по размеру

Очень малые БПЛА

К классу очень малых БПЛА относятся БПЛА с размерами от размера крупного насекомого до 30-50 см в длину. Насекомоподобные БПЛА с машущими или вращающимися крыльями являются популярной микроконструкцией. Они чрезвычайно малы по размеру, очень легки и могут использоваться для шпионажа и биологической войны. Более крупные используют обычную конфигурацию самолета. Выбор между машущими или вращающимися крыльями зависит от желаемой маневренности. Конструкции на основе машущих крыльев позволяют садиться и приземляться на небольшие поверхности. Примерами очень малых БПЛА являются израильский *IAI Malat Mosquito* (с размахом крыльев 35 см и продолжительностью полета 40 минут), американский *Aurora Flight Sciences Skate* (с размахом крыльев 60 см и длиной 33 см), австралийский *Cyber Technology CyberQuad Mini* (с квадратом 42x42 см) и их последняя модель *CyberQuad Maxi*. См. рисунок 1 ниже.

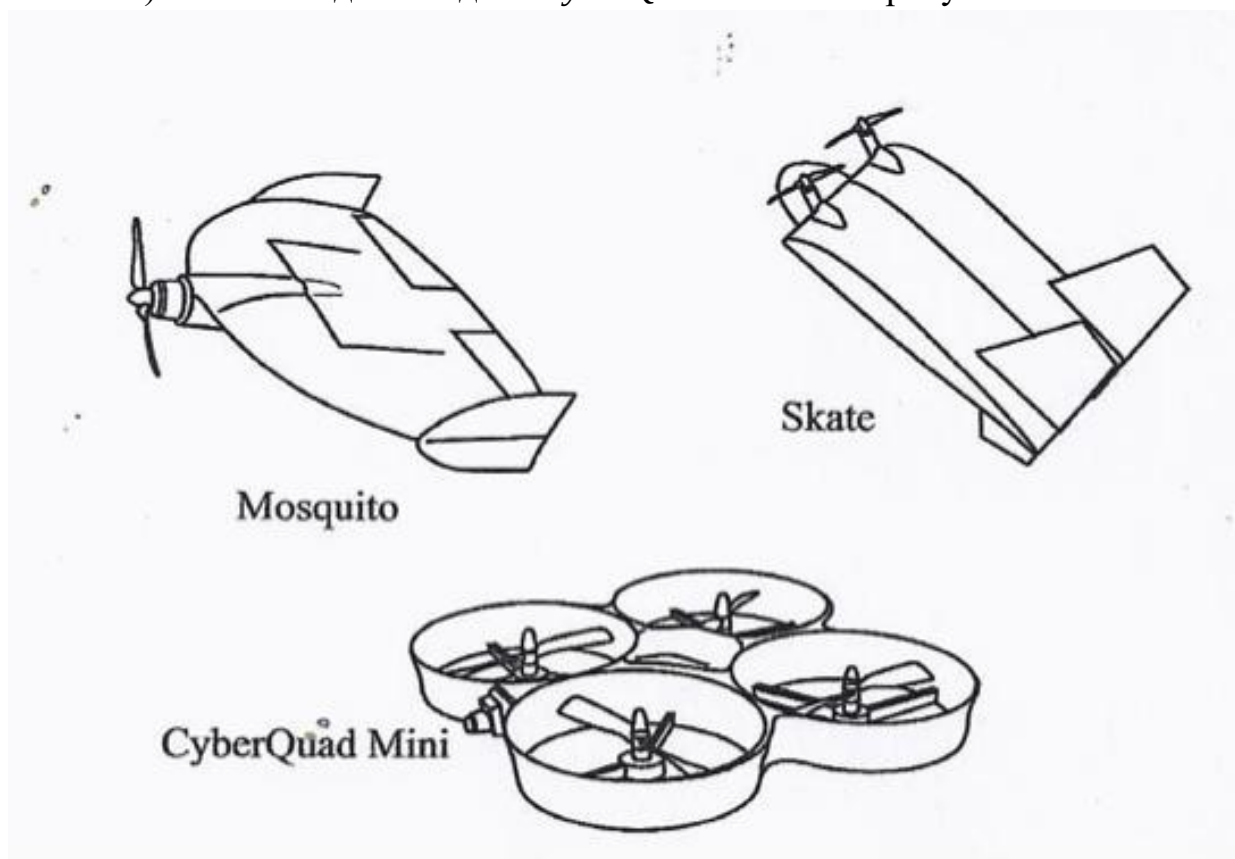


Рисунок 1. Примеры очень маленьких БПЛА

Малые БПЛА

Класс малых БПЛА (иногда также называемых мини-БПЛА) относится к БПЛА, у которых хотя бы одно измерение больше 50 см и не больше 2 метров. Многие из конструкций в этой категории основаны на модели с фиксированным крылом, и большинство из них запускаются вручную, подбрасывая

их в воздух, как показано на рисунке 2. Примерами представителей этого класса малых БПЛА являются:

- *RQ-11 Raven* длиной 1 метр от *US Aero Vironment* с размахом крыльев 1,4 м;
- Турецкий *Bayraktar* (рисунок 5), который весит около 5 кг и имеет дальность передачи данных 20 км;
- Армейский *RQ-7 Shadow* (рисунок 3);
- *The AiRanger™* от *American Aerospace*, показанный на рисунке 4. *AiRanger™* – это кроссоверный БПЛА между малой и средней системой.

Некоторые БПЛА этого класса основаны на конструкции винтокрылого аппарата.



Рисунок 2. Малый БПЛА, запускаемый вручную



Рисунок 3. RQ-7 Shadow армии США. RQ-7B, или Shadow 200,
- это тип беспилотного летательного аппарата, которым управляет VMU-3



Рисунок 4. AiRanger™

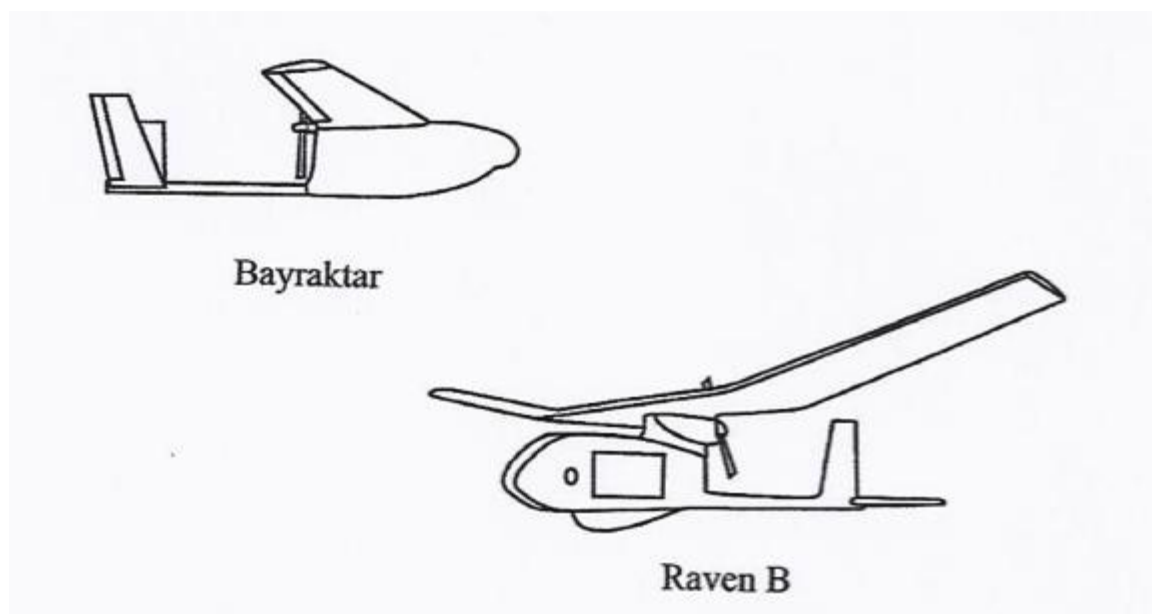


Рисунок 5. Примеры малых БПЛА

Средние БПЛА

К классу средних БПЛА относятся БПЛА, которые слишком тяжелы для переноски одним человеком, но все еще меньше легкого самолета. Обычно они имеют размах крыльев около 5–10 м и могут нести полезную нагрузку от 100 до 200 кг. Примерами средних БПЛА с фиксированным крылом являются (рисунок 6 ниже) израильско-американский *Hunter* и британский *Watchkeeper*. В прошлом использовались и другие бренды, такие как американский *Boeing Eagle Eye*, *RQ-2 Pioneer*, *BAE systems Skyeeye R4E* и *RQ-5A Hunter*. У *Hunter* размах крыльев составляет 10,2 м, а длина – 6,9 м. Он весит около 885 кг на взлете. *RS-20* от *American Aerospace* – еще один пример кроссовера БПЛА, который охватывает характеристики малого и среднего БПЛА. В задании по чтению можно найти много других средних БПЛА. Также есть ряд роторных БПЛА среднего размера.

Большие БПЛА

Класс больших БПЛА относится к большим БПЛА, используемым в основном для боевых операций военными. Примерами таких больших БПЛА являются американские *General Atomics Predator A* и *B* и американский *Northrop Grumman Global Hawk* (рисунки 7 и 8).

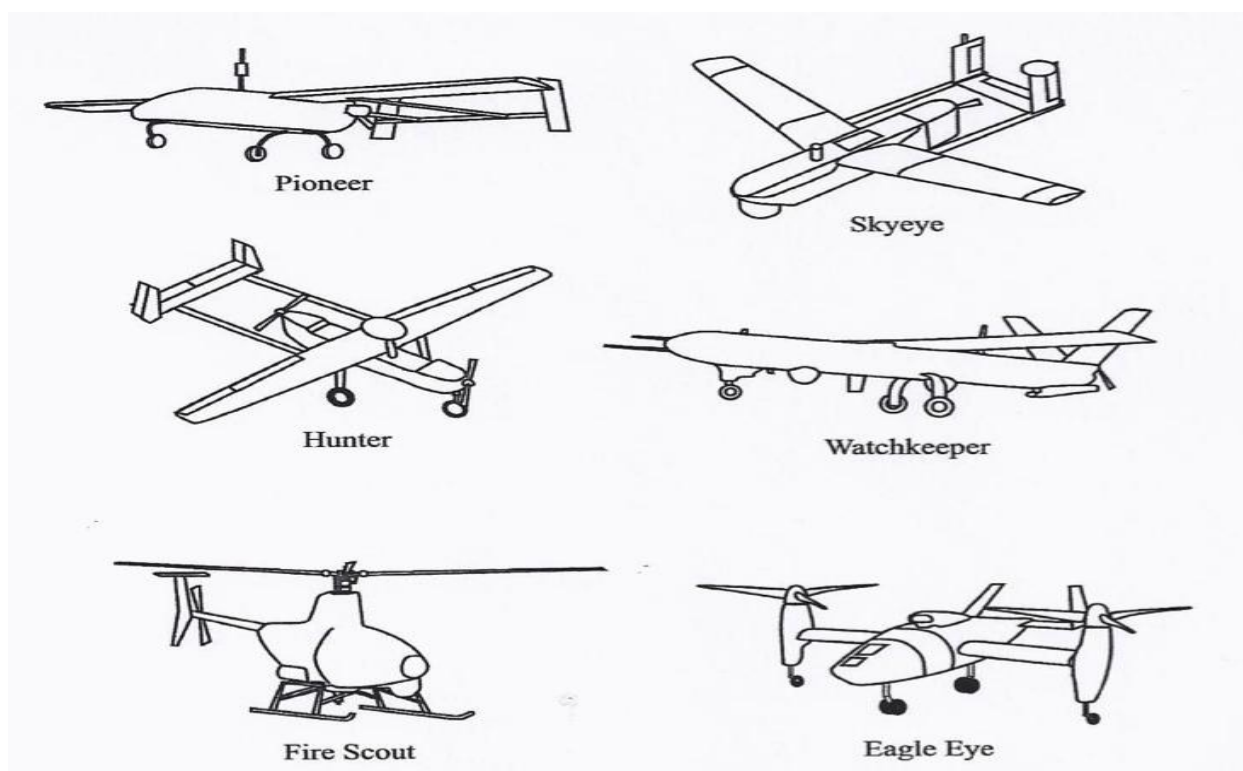


Рисунок 6. Примеры средних БПЛА

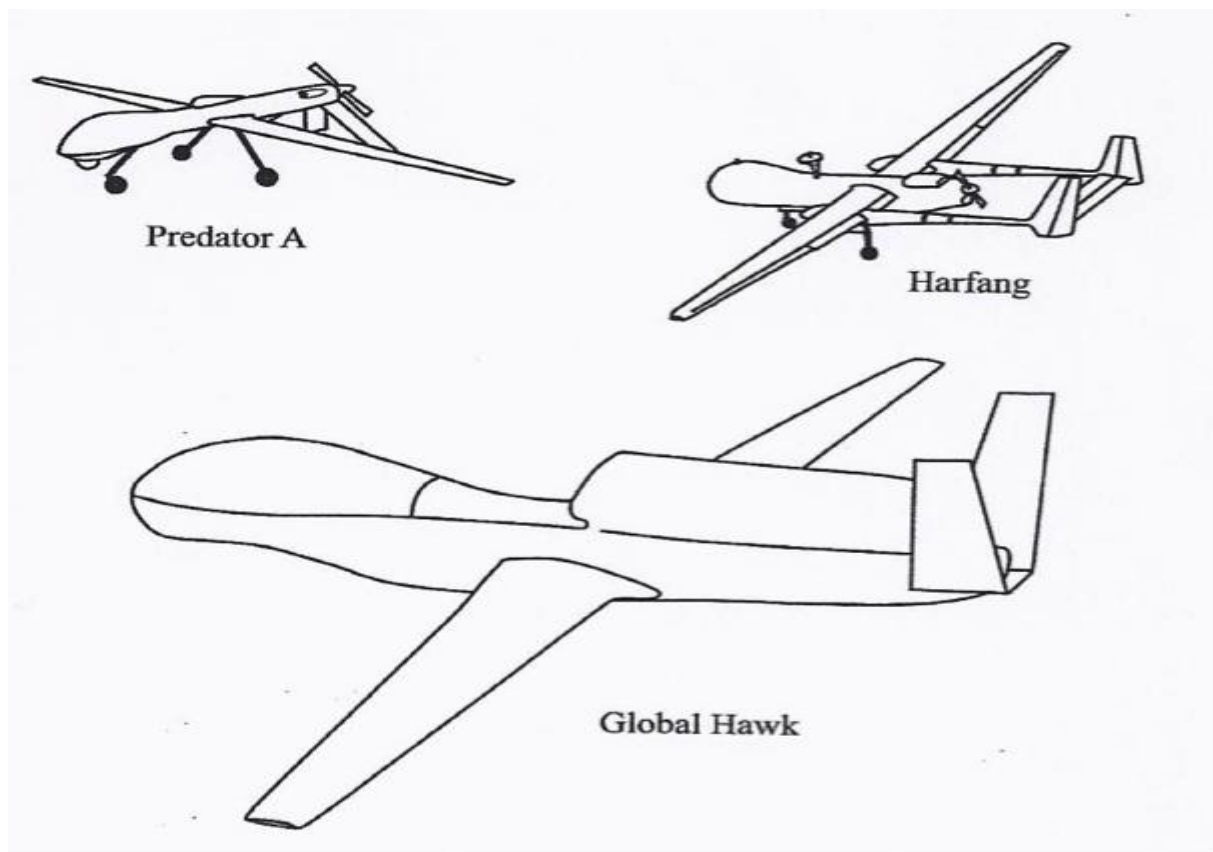


Рисунок 7. Примеры больших БПЛА



Рисунок 8. *Global Hawk* от NASA

Классификация по дальности и выносливости

БПЛА ближнего действия

В этот класс входят БПЛА с дальностью полета 50 км и временем автономной работы от 1 до 6 часов. Обычно они используются для задач разведки и наблюдения.

БПЛА среднего действия

К классу среднего действия относятся БПЛА со сверхвысокой скоростью и радиусом действия 650 км. Они также используются для целей разведки и наблюдения, в дополнение к сбору метеорологических данных.

БПЛА дальнего действия

К классу выносливости относятся БПЛА с выносливостью 36 часов и радиусом действия 300 км. Этот класс БПЛА может работать на высоте 30 000 футов. Они также используются для целей разведки и наблюдения.

Список использованных источников

1. Емельянов, Д.А. Фильтрация сетевого контента в образовательных учреждениях / Д.А. Емельянов // М. : Педагогическое образование в России – 2018. – URL: <https://cyberleninka.ru/article/n/filtratsiya-setevogo-kontenta-v-obrazovatelnyh-uchrezhdeniyah/viewer>
2. Михеева, Т.И. Интеллектуальная транспортная геоинформационная система ITSGIS. Ядро / Т.И. Михеева, С.В. Михеев, О.К. Головнин, А.В. Сидоров, Е.А. Савинов. – Самара : Интелтранс, 2016. – Т.1. – 171 с.
3. Воронина, Е.Е. Фотовидеофиксация как средство профилактики дорожно-транспортных происшествий / Е.Е. Воронина // Вестник НЦБЖД. – 2019. – № 2(40). – С. 73-79.
4. Mikheeva, T. Recognition of Urban Transport Infrastructure Objects Via Hyperspectral Images / O. Saprykin, A. Fedoseev, T. Mikheeva // In Proceedings of the 2nd International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS-2016). Rome, Italy: SCITEPRESS, 2016. – P. 203-208.
5. Минниханов, Р.Н. Применение данных ИТС систем для моделирования эффективной транспортной среды в Республике Татарстан / Р.Н. Минниханов, С.С. Степанова, В.А. Решетов, А.З. Махмутова // Безопасность дорожного движения. – 2021. – № 3. – С. 8-12.
6. Михеев, С.В. Интеллектуальная транспортная геоинформационная система ITSGIS / Т.И. Михеева, С.В. Михеев, О.К. Головнин // Современные проблемы безопасности жизнедеятельности: интеллектуальные транспортные системы : материалы IV Международной науч-

- но-практической конференции (Казань, 25–26 февраля 2016 г.). – Казань : ГБУ «Научный центр безопасности жизнедеятельности», 2016. – С. 362–368. – ISBN 978-5-85247-837-5.
7. Михеева, Т.И. Метод синтеза системы зонального сетевидного управления транспортными процессами / Т.И. Михеева, С.В. Михеев, О.К. Головнин // Известия Самарского научного центра Российской академии наук. – 2016. – Т. 18, № 4 (4). – С. 799-807.
 8. Михеева, Т.И. Системный анализ объектов транспортной инфраструктуры в геоинформационной среде / Т.И. Михеева // Программные продукты и системы. – 2018. – № 1. – С. 12-18.
 9. Головнин, О.К. Применение паттерна «Рабочий поток» в геоинформационной составляющей интеллектуальной транспортной системы / О.К. Головнин // Математика. Компьютер. Образование : тезисы докладов двадцатой международной конференции. – Пущино, 2013. – Режим доступа : <http://www.mce.su/rus/archive/abstracts/mce20/sect101361/doc174037>.
 10. Михеев, С.В. Модели наследования в системе управления дорожным движением / Т.И. Михеева, С.В. Михеев // Информационные технологии. – 2001. – № 7. – С. 50-54.
 11. Турушев, Т.К. Архитектурные паттерны проектирования системы – MVC, MVP, MVVM / Т.К. Турушев // Инновационный потенциал развития науки в современном мире: достижения и инновации : Сборник научных статей по материалам VI Международной научно-практической конференции, Уфа, 22 октября 2021 года. – Уфа: Общество с ограниченной ответственностью «Научно-издательский центр «Вестник науки», 2021. – С. 16–21.

Mikheeva T.I., Klepikov N.M.

CLASSIFICATION OF UNMANNED AERIAL VEHICLES

*Samara, Samara University named after academician S.P. Korolev
IntelTrans*

The article discusses the classification of unmanned aerial vehicles. There are varieties according to which it is desirable to use unmanned aerial vehicles to solve the problem of object recognition.

Keywords: unmanned aerial vehicle, classification, object detection, orthoimage, electronic map.

*Михеева Т.И.***СОВРЕМЕННЫЕ МЕТОДЫ МИКРОМОДЕЛИРОВАНИЯ ДВИЖЕНИЯ
ТРАНСПОРТНЫХ СРЕДСТВ В «ITS GIS»: РАЗРАБОТКИ И
ПРОЕКТИРОВАНИЯ СЛОЖНЫХ ПРОГРАММНЫХ КОМПЛЕКСОВ***Самара, Самарский университет имени академика С.П. Королёва
ИнтелТранс*

В данной статье приводится во время курсового проекта разработана система, которая позволяет в автоматическом или ручном режимах моделировать улично-дорожную сеть (УДС) с заданными параметрами моделирования. Схема смоделированной УДС может храниться в файлах с последующей загрузкой в программу. Раскрываются методы и средства, которые были применены при реализации полученных решений в системе ITS GIS.

Ключевые слова: моделирование, улично-дорожная сеть, интенсивность, геоинформационная система, системы аналоги, перекресток, дорога.

Введение

Современное общество нуждается в постоянном увеличении объема транспортного сообщения, повышении его надежности, безопасности и качества. Это требует увеличения затрат на улучшение инфраструктуры транспортной сети, превращения ее в гибкую, высоко управляемую логистическую систему. При этом риск инвестиций значительно возрастает, если не учитывать закономерности развития транспортной сети, распределение загрузки ее участков. Игнорирование этих закономерностей приводит к частому образованию транспортных пробок, перегрузке/недогрузке отдельных линий и узлов сети, повышению уровня аварийности, экологическому ущербу.

Для поиска эффективных стратегий управления транспортными потоками в мегаполисе, оптимальных решений по проектированию улично-дорожной сети и организации дорожного движения необходимо учитывать широкий спектр характеристик транспортного потока, закономерности влияния внешних и внутренних факторов на динамические характеристики смешанного транспортного потока. Во время выполнения курсового проекта необходимо разработать автоматизированную систему для моделирования транспортного потока, которая позволит составлять карту дороги, моделировать движение машин.

Во время разработки система будет использовать технологию *RAD (Rapid Application Development* – быстрая разработка приложений), которая служит для создания средств разработки программных. Особое внимание уделяется скорости и удобству программирования, созданию технологического процесса, позволяющего программисту максимально быстро создавать компьютерные программы. *RAD* – это жизненный цикл процесса проектирования, созданный для достижения более высокой скорости разработки и качества *ITSGIS*, чем это возможно при традиционном подходе к проектированию.

Во время проектирования будет использоваться методология *UML (Unified Modeling Language* – унифицированный язык моделирования), которая служит для графического описания объектного моделирования в области разработки программного обеспечения. *UML* является языком широкого профиля, это – открытый стандарт, использующий графические обозначения для создания абстрактной модели системы, называемой *UML*-моделью. *UML* – создан для определения, визуализации, проектирования и документирования, в основном, программных систем. *UML* не является языком программирования, но на основании *UML*-моделей возможна генерация кода.

Описание предметной интеллектуальная транспортная геоинформационная система «ITSGIS»

Моделирование – исследование объектов познания на их моделях, построение и изучение моделей реально существующих объектов, процессов или явлений с целью получения объяснений этих явлений, а также для предсказания явлений, интересующих исследователя.

История возникновения моделирования движения машин

Первостепенной задачей, послужившей развитию моделирования транспортных потоков, стал анализ пропускной способности магистралей и пересечений. Под пропускной способностью понимают максимально возможное число автомобилей, которое может пройти через сечение дороги за единицу времени. В специальной литературе встречаются такие модификации понятия пропускной способности, как теоретическая, номинальная, эффективная, собственная, практическая, фактическая и другие. В настоящее время пропускная способность является важнейшим критерием оценки качества функционирования путей сообщения.

Первая макроскопическая модель, в которой движение транспортного потока рассматривалось с позиций механики сплошной среды. Они показали, что методы описания процессов переноса в сплошных средах могут быть использованы для моделирования заторов.

В 60–70-е годы вновь возник интерес к исследованию транспортных систем. Эта заинтересованность проявилась в том числе, в финансировании многочисленных контрактов, обращении к авторитетным ученым – специалистам в области математики, физики, процессов управления. Сегодня имеется обширная литература по изучению и моделированию автотранспортных потоков. Несколько академических журналов посвящены исключительно динамике автомобильного движения. Наиболее крупными являются *Transportation Research*, *Transportation Science*, *Mathematical Computer Simulation*, *Operation Research*, *Automatica*, *Physical Review E*, *Physical Reports*. Количество публикуемых статей исчисляется сотнями.

В конце 80-х начале 90-х в США проблемы исследования транспортных систем были возведены в ранг проблем национальной безопасности. К решению этой задачи были привлечены лучшие «физические умы» и компьютерная техника.

Основные понятия предметной области

Для моделирования дорожного движения исключительно важно понимать основные понятия предметной области.

Дорога – обустроенная или приспособленная и используемая для движения транспортных средств полоса земли либо поверхность искусственного сооружения. Дорога включает в себя одну или несколько проезжих частей, а также трамвайные пути, тротуары, обочины и разделительные полосы при их наличии.

Перекрёсток – это место пересечения, примыкания или разветвления дорог на одном уровне, ограниченное воображаемыми линиями, соединяющими соответственно противоположные, наиболее удаленные от центра перекрёстка начала закруглений проезжих частей (Рисунок 1).

Светофор – оптическое устройство, несущее световую информацию. Светофор предназначен для регулирования движения механических транспортных средств, а также пешеходов на пешеходных переходах и иных участников дорожного движения, поездов железной дороги и метрополитена, речных и морских судов, трамваев, троллейбусов, автобусов и иного транспорта. В странах СНГ светофор на городских улицах, является муниципальной собственностью (Рисунок 2).



Рисунок 1. Пример перекрестка



Рисунок 2. Пример светофора

Автомобиль – механическое транспортное средство, используемое обычно для перевозки по дорогам людей или грузов, или для буксировки по дорогам транспортных средств, используемых для перевозки людей или грузов.

Транспортная система – транспортная инфраструктура, транспортные предприятия, транспортные средства и управление в совокупности. Единая транспортная система обеспечивает согласованное развитие и функционирование всех видов транспорта с целью максимального удовлетворения транспортных потребностей при минимальных затратах.

Улично-дорожная сеть (УДС) – совокупность городских дорог, улиц, проездов, включая основную проезжую часть, газоны, тротуары, наружное освещение и другие элементы благоустройства, а также дорожные покрытия инженерных сооружений (мостов, путепроводов) (Рисунок 3).

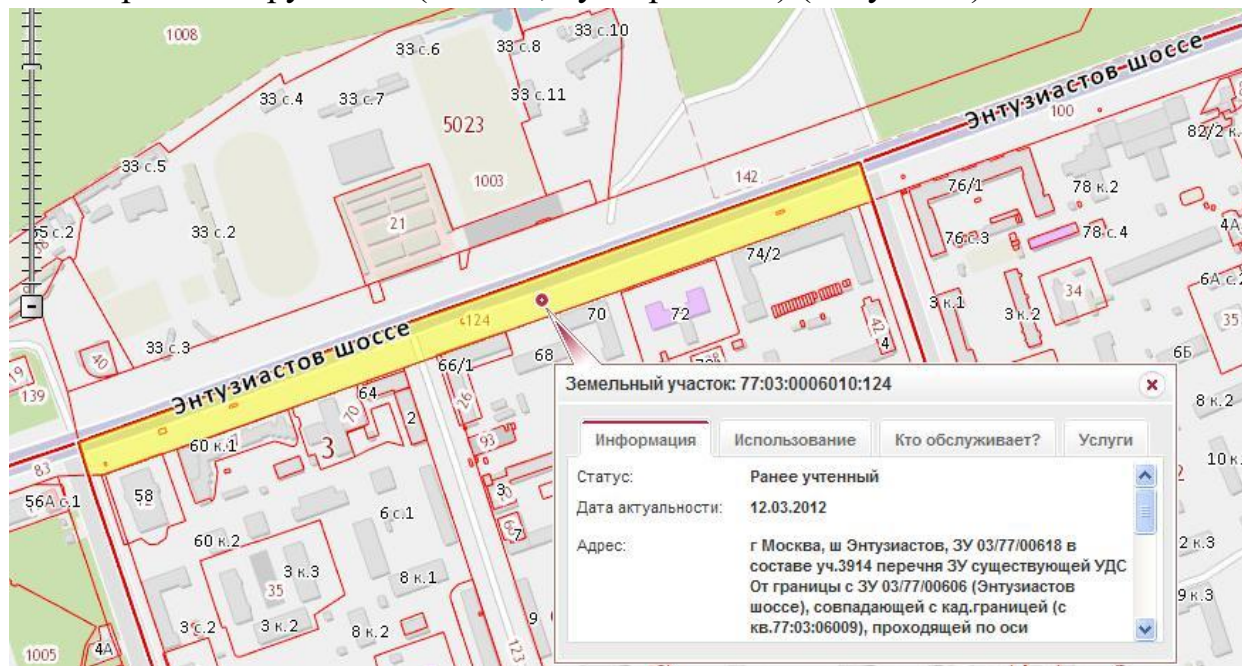


Рисунок 3. Пример улично-дорожная сеть

Моделирование УДС

Моделирование – исследование объектов познания на их моделях; построение и изучение моделей реально существующих объектов, процессов или явлений с целью получения объяснений этих явлений, а также для предсказания явлений, интересующих исследователя.

Процесс моделирования включает элементы:

- субъект (исследователь),
- объект исследования,
- модель, определяющую (отражающую) отношения познающего субъекта и познаваемого объекта.

Первый этап построения модели предполагает наличие некоторых знаний об объекте-оригинале. Познавательные возможности модели обуславливаются тем, что модель отображает (воспроизводит, имитирует) какие-либо существенные черты объекта-оригинала. Вопрос о необходимой и достаточной мере сходства оригинала и модели требует конкретного анализа. Очевидно, модель утрачивает свой смысл как в случае тождества с оригиналом (тогда она перестает быть моделью), так и в случае чрезмерного во всех существенных отношениях отличия от оригинала. Таким образом, изучение одних сторон моделируемого объекта осуществляется ценой отказа от исследования других сторон. Поэтому любая модель замещает оригинал лишь в стро-

го ограниченном смысле. Из этого следует, что для одного объекта может быть построено несколько «специализированных» моделей, концентрирующих внимание на определенных сторонах исследуемого объекта или же характеризующих объект с разной степенью детализации.

На втором этапе модель выступает как самостоятельный объект исследования. Одной из форм такого исследования является проведение «модельных» экспериментов, при которых сознательно изменяются условия функционирования модели и систематизируются данные о её «поведении». Конечным результатом этого этапа является множество (совокупность) знаний о модели.

На третьем этапе осуществляется перенос знаний с модели на оригинал – формирование множества знаний. Одновременно происходит переход с «языка» модели на «язык» оригинала. Процесс переноса знаний проводится по определенным правилам. Знания о модели должны быть скорректированы с учетом тех свойств объекта-оригинала, которые не нашли отражения или были изменены при построении модели.

Сейчас трудно указать область человеческой деятельности, где не применялось бы моделирование. Разработаны, например, модели производства автомобилей, выращивания пшеницы, функционирования отдельных органов человека, жизнедеятельности Азовского моря, последствий атомной войны. В перспективе для каждой системы могут быть созданы свои модели, перед реализацией каждого технического или организационного проекта должно проводиться моделирование.

Случайные потоки широко используются в качестве математических моделей в задачах, связанных с исследованием систем массового обслуживания, в задачах приема импульсных сигналов, в задачах надежности и т. п. Они являются специфичным классом случайных процессов, определяют случайные моменты времени, в которые происходят некоторые события.

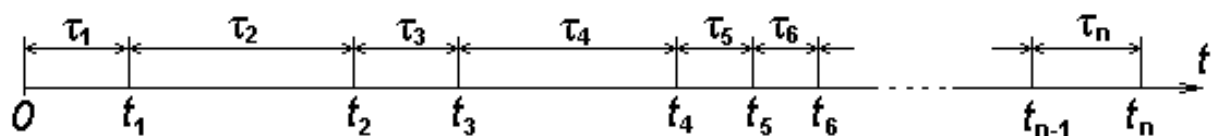


Рисунок 4. Случайный поток событий

В общем случае случайные потоки задаются с помощью многомерной плотности распределения вероятностей интервалов времени между двумя соседними событиями. При таком задании случайных потоков моделирование их сводится к формированию реализаций случайных векторов с законом распределения, для чего могут быть использованы известные методы.

Детерминированные процессы можно рассматривать как частный случай стохастических процессов, плотность распределения которых представляет собой импульсную функцию Дирака, т. е. δ -функцию: $p(x) = \delta(x)$. При этом условные математические ожидания, входящие в критерий оптимальности и ограничения превращаются просто в детерминированные функции, не зависящие от случайного вектора x . В этом случае временные интервалы $\tau_1 = \tau_2 = \tau_3 \dots \tau_n = \text{const}$ для всех событий в потоке.

Стохастические процессы отличаются друг от друга и, в частности, от детерминированных процессов видом вероятностных характеристик – плотностей распределения временных интервалов $\tau_1, \tau_2 \dots \tau_n$.

Метод моделирования случайных событий характеризуется законом распределения временных интервалов между событиями $\tau_1, \tau_2 \dots \tau_n$.

Поток Пуассона – вероятностное распределение дискретного типа, моделирует случайную величину, представляющую собой число событий, произошедших за фиксированное время, при условии, что данные события происходят с некоторой фиксированной средней интенсивностью и независимо друг от друга.

Экспоненциальным распределением называется частный случай гамма-распределения с параметрами $a = 1$; $b = \lambda > 0$, то есть то есть плотность вероятности в этом случае $f(x) = \begin{cases} 0, & x \leq 0; \\ \lambda e^{-\lambda x}, & x > 0. \end{cases}$

Используя свойства два плотности распределения, можно найти функцию распределения $F(x)$ экспоненциального закона:

$$F(x) = \begin{cases} 0, & x < 0; \\ 1 - e^{-\lambda x}, & x \geq 0. \end{cases}$$

Основные характеристики (математическое ожидание и дисперсия) случайной величины X , распределённой по экспоненциальному, имеют вид

$$M(X) = \frac{1}{\lambda}; \quad D[X] = \frac{1}{\lambda^2}.$$

Статистический смысл параметра λ состоит в следующем: λ есть среднее $\frac{1}{\lambda}$ число событий на единицу времени, то есть $\frac{1}{\lambda}$ средний промежуток времени между двумя последовательными событиями.

Методы моделирования УДС

Необходимость получения случайных чисел, подчиняющихся распределению определенного вида при моделировании связана со случайным характером изучаемых процессов. Для получения таких чисел служит равномерно распределенная последовательность случайных чисел на интервале $[0, 1)$, которая используется для получения случайных чисел любого типа распределе-

ния. В программное обеспечение ЭВМ включены специальные программы – датчики случайных чисел, обращение к которым приводит к выдаче случайного значения в интервале.

В основе моделирования случайных величин с произвольными законами распределения вероятностей лежит, как правило, метод обратной функции.

В этом методе используется следующая теорема. По определению, $F(y)$ является функцией распределения случайной величины Y .

Теорема может быть проиллюстрирована графиками, представленными на рисунке 5.

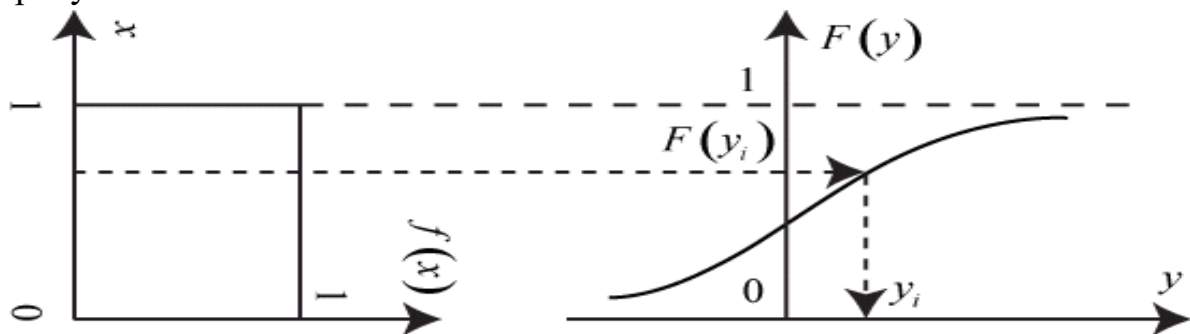


Рисунок 5. Случайный поток событий

Равномерно распределенное случайное число x_i смоделировано. Нужно найти неизвестное y_i , находящееся в верхнем пределе интегрирования. Таким образом, для экспоненциального закона распределения обратная функция будет иметь следующий вид: $y_i = 1 - \ln(x_i)$.

Обзор существующих систем-аналогов

Основные программы аналоги являются средства для моделирования динамических систем. Рассмотрим некоторые из них.

Система Visum

VISUM представляет собой обширную, гибкую программу для моделирования транспортных потоков, расчета спроса на транспорт (матрицы корреспонденций для общественного и индивидуального транспорта), анализа транспортной сети, расчет себестоимости общественного транспорта и прогноза запланированных мероприятий и их последствий.

Область применения транспортных моделей включает следующий список задач:

- хранение обширной базы данных транспортных и социально-экономических показателей;
- расчет объемов существующих транспортных потоков;
- расчет транспортной потребности городов и регионов;
- математическое моделирование транспортных потоков.

Также существуют различные оценки. Например, оценка различных транспортных ситуаций и вариантов развития транспортной инфраструктуры по заданной системе показателей.

Оценка работы транспортной сети в целом по разработанной системе показателей качества. Такими показателями могут быть, например, транспортная обеспеченность территории, оценки транспортной доступности территорий для различных групп населения и прочее.

Систематизация и наглядное представление данных по транспортной системе города, региона, страны (например, для визуальной оценки и разработки предложений).

На рисунках 6, 7 показан внешний вид приложения.



Рисунок 6. Экранная форма программы «Visum»

с пешеходным переходом

Система AnyLogic

AnyLogic — программное обеспечение для имитационного моделирования, разработанное российской компанией *The AnyLogic Company* (бывшая «Экс Джей Текнолоджис», англ. *XJ Technologies*). Инструмент обладает современным графическим интерфейсом и позволяет использовать язык *Java* для разработки моделей. Уникальность, гибкость и мощность языка моделирования, предоставляемого *AnyLogic*, позволяет учесть любой аспект моделируемой системы с любым уровнем детализации.



Рисунок 7. Экранная форма программы «Visum» с перекрёстком

AnyLogic включает в себя графический язык моделирования, а также позволяет пользователю расширять созданные модели с помощью языка *Java*. Интеграция компилятора *Java* в *AnyLogic* предоставляет более широкие возможности при создании моделей, а также создание *Java* апплетов, которые могут быть открыты любым браузером. Эти апплеты позволяют легко размещать модели *AnyLogic* на веб-сайтах. В дополнение к *Java*-апплетам, *AnyLogic Professional* поддерживает создание *Java*-приложений, в этом случае пользователь может запустить модель без инсталляции *AnyLogic*.

Графический интерфейс *AnyLogic*, инструменты и библиотеки позволяют быстро создавать модели для широко спектра задач от моделирования производства, логистики, бизнес-процессов до стратегических моделей развития компании и рынков:

- проектирование агентных, системно-динамические, дискретно-событийные и многоподходные модели, используя только один инструмент;
- *AnyLogic* поддерживает как дискретный, так и непрерывный подход в пределах одной модели;
- *Java* платформа инструмента *AnyLogic* предоставляет безграничную расширяемость моделей за счет программирования на *Java*, создания пользовательских библиотек и работы с базами данных;
- богатый набор функций распределения позволяет создавать сложные стохастические модели;
- сильная экспериментальная база, встроенная поддержка моделирования Монте-Карло и передовых форм оптимизации дает большое разнообразие подходов моделирования.

На рисунках 8, 9 представлена экранная форма программы.

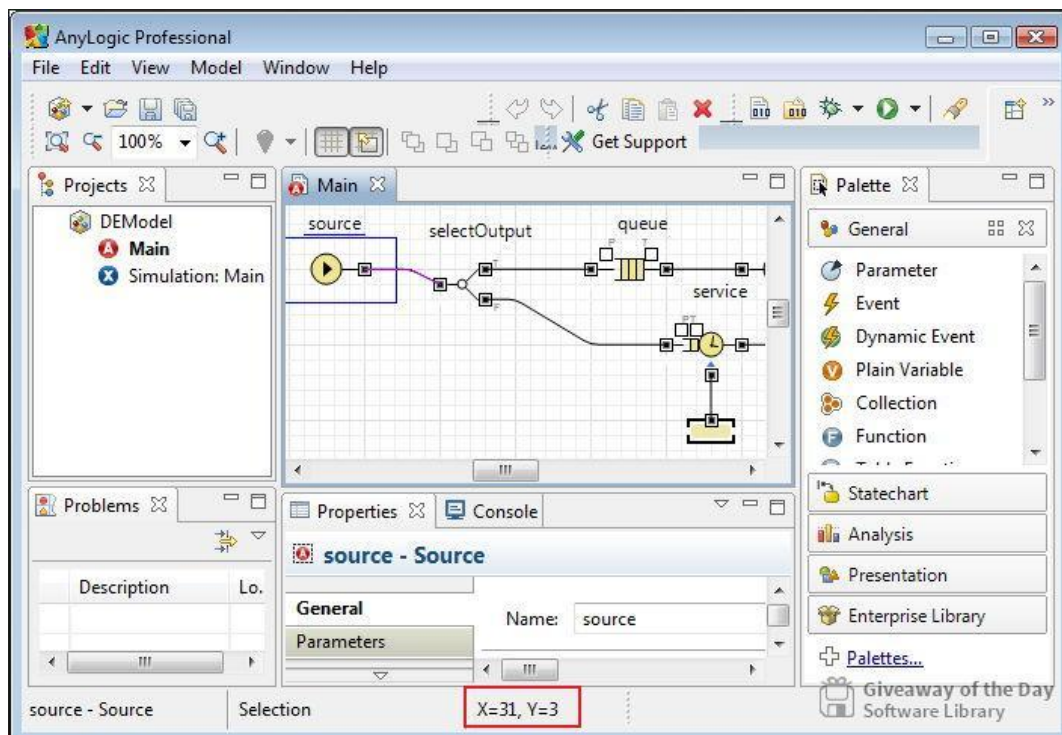


Рисунок 8. Экранная форма программы «AnyLogic»

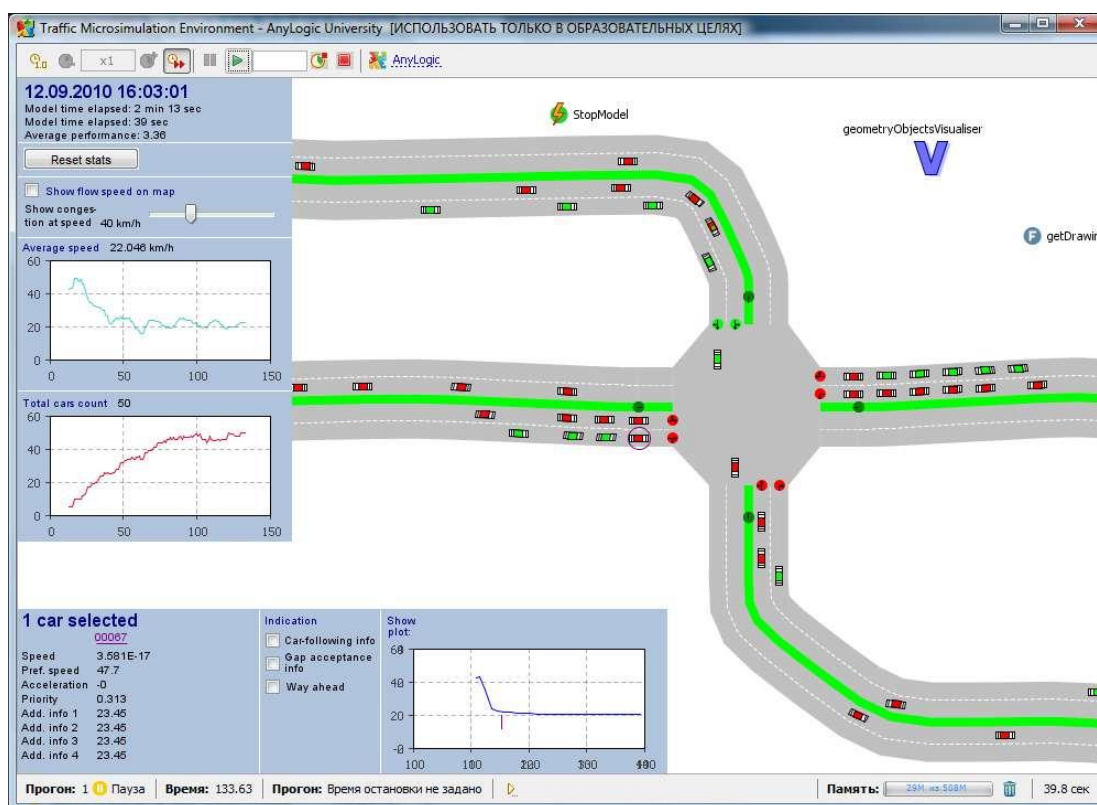


Рисунок 9. Экранная форма программы «AnyLogic»

Постановка задачи

Перед авторами поставлена задача разработать автоматизированную систему моделирования дорожного движения по технологии *RAD* и методологии *UML*.

В системе предусмотрено два варианта построения УДС *ITSGIS*: автоматический, ручной.

При автоматическом режиме построения *ITSGIS* система случайным образом должна расставлять перекрестки, соединяя их дорогами.

При ручном режиме пользователь может сам расставлять перекрестки, соединять их дорогами.

Система в обоих режимах автоматически должна добавлять автомобили в соответствии с заданным количеством, на каждом перекрестке автоматически должна добавлять необходимое количество светофоров, с длительностью зеленого и красного света равным 140 тиков модельного времени.

В процессе моделирования пользователь должен иметь возможность изменять следующие параметры:

- задание количества машин (от 1 до 200);
- задание модельного времени (от 0.1с до 10с);
- задание масштаба (от 0.1 до 2).

Система должна отображать УДС в соответствии с заданными параметрами. При необходимости пользователь должен иметь возможность сохранить структуру УДС в файл с последующей возможностью загрузить сохраненную копию.

В системе также должна быть обеспечена возможность получения справочной информации, как о самой системе, так и предоставляемых ею возможностях.

Таким образом, система должна выполнять следующие функции *ITSGIS*:

1) настройка параметров моделирования:

- 1 задание количество машин;
- 2 задание модельного времени.

2) автоматическая генерация УДС;

3) ручное составление УДС:

- 1 добавление дороги;
- 2 удаление дороги;
- 3 добавление перекрестка;
- 4 удаление перекрестка;
- 5 добавление истока/стока;
- 6 удаление истока/стока;
- 7 автоматическое добавление светофоров;
- 8 автоматическое удаление светофоров.

4) визуализация отображение процесса моделирования;

5) сохранение УДС в файл;

6) загрузка УДС из файла;

7) выдача справочной информации о системе.

Проектирование системы

Система – множество элементов, находящихся в отношениях и связях друг с другом, которое образует определённую целостность, единство. Отличительным свойством системы является ее целостность: комплекс объектов, рассматриваемых в качестве системы, должен обладать общими свойствами и поведением.

Структурная схема системы

Система должна представлять собой совокупность элементов, находящихся между собой в определенной зависимости и составляющих некоторое единство, направленное на достижение определенной цели. Система может являться элементом другой системы более высокого порядка и включать в себя системы более низкого порядка. То есть систему можно рассматривать как набор подсистем, организованных для достижения определенной цели и описанных с помощью набора моделей, а подсистему – как группу элементов, часть которых составляет спецификацию поведения, представленного другими ее составляющими.

Структурная схема разрабатываемой системы приведена на рисунке 10.

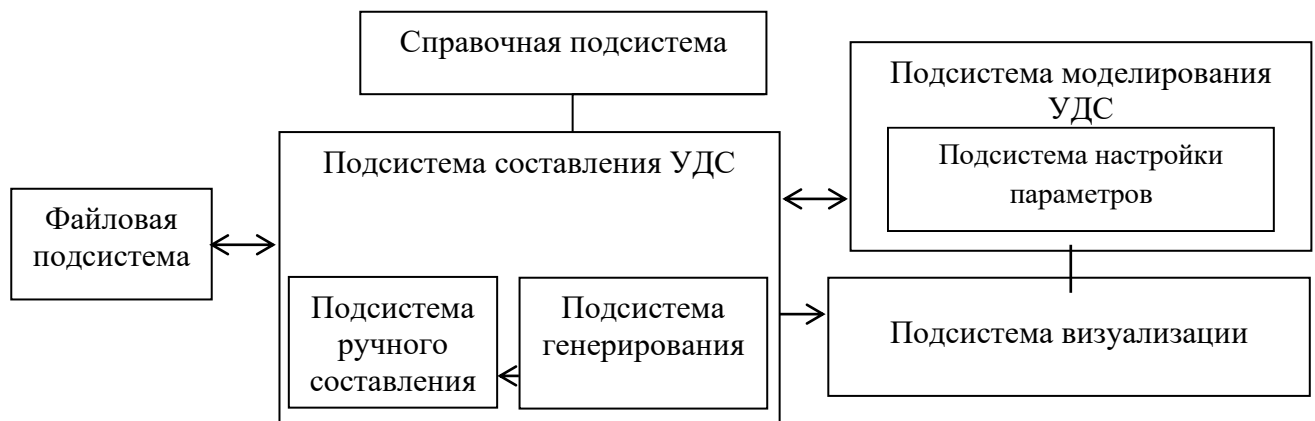


Рисунок 10. Структурная схема системы

В состав системы входят следующие подсистемы:

1) подсистема составления УДС *ITSGIS*, в состав которой входят:

- подсистема ручного составления, которая отвечает за ручное построение дороги, её удаления, связывания с перекрестками и её перемещение по сетке;
- подсистема генерирования, которая отвечает за автоматическое составление дорожной карты с перекрестками и дорогами;

2) подсистема визуализации, которая отвечает за корректное отображение карты мира у пользователя;

3) подсистема моделирования УДС, в состав которой входит подсистема настройки параметров, которая отвечает за ввод (выбор) значений параметров модельного мира и проверку корректности этих значений;

4) файловая подсистема, которая отвечает за хранение УДС в файле заранее определенного типа;

5) справочная подсистема, которая содержит сведения о системе (руководство пользователю) и ее об ее разработчиках.

Проектирование интерфейса пользователя

Интерфейс (от англ. *interface* – поверхность раздела, перегородка) – совокупность средств и методов взаимодействия между элементами системы. Интерфейс является основной и наиболее важной частью программы. Поскольку пользователь управляет программой и данными именно через интерфейс, уделяем большое внимание понятности и простоте интерфейсов в разработанных нами программах.

Для выполнения начальной фазы разработки необходимо погрузиться целиком в задачи пользователей и создать бумажный прототип навигационной модели. Навигационная модель показывает, как необходимо распределять функции или задачи между окнами вашей программы, она определяет, как пользователи смогут перемещаться как между различными задачами, так и внутри отдельной задачи. На рисунке 11 приведен прототип интерфейса пользователя.

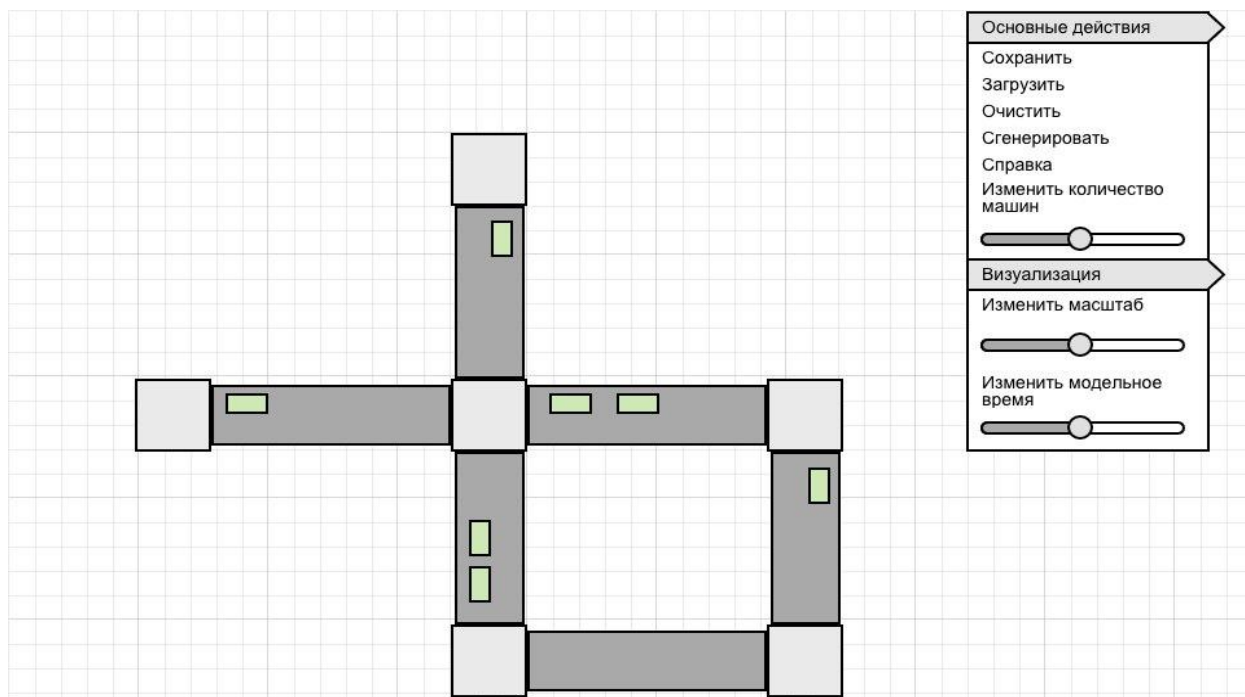


Рисунок 11. Прототип интерфейса пользователя

При нажатии на кнопку «Справка» будет открываться справочное окно, содержащее информацию об авторах и документацию. Данное окно изобра-

жено на рисунке 12. Вернуться в главное меню можно будет по нажатию кнопки «Главное меню».

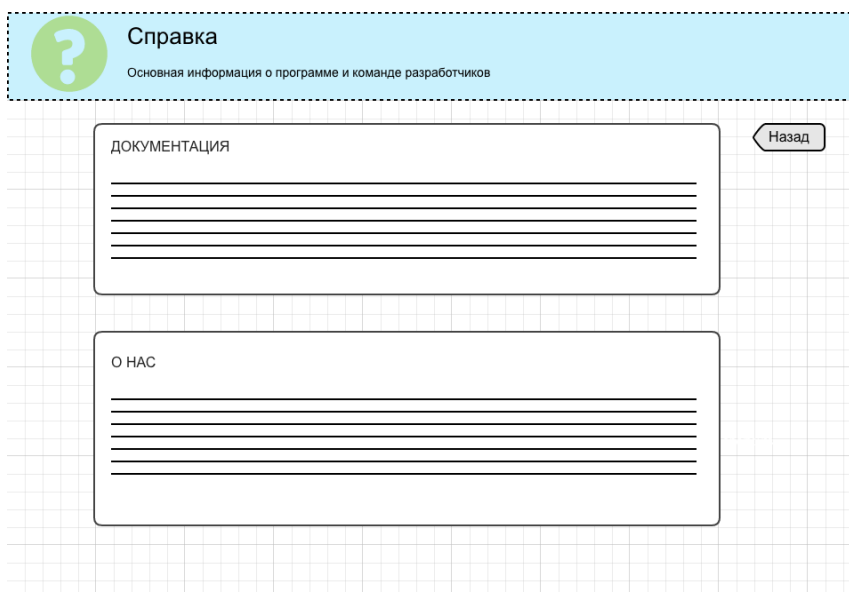


Рисунок 12. Прототип интерфейса справка

Информационно-логический проект системы

Модель — это упрощенное представление реальности, по существу — это «чертеж» системы: в нее может входить как детальный план, так и более абстрактное представление системы «с высоты птичьего полета». Хорошая модель всегда включает элементы, которые существенно влияют на результат, и не включает те, которые малозначимы на данном уровне абстракции.

Моделирование позволяет решить четыре различные задачи:

- 1 визуализировать систему в ее текущем или желательном для нас состоянии;
- 2 описать структуру или поведение системы;
- 3 получить шаблон, позволяющий сконструировать систему;
- 4 документировать принимаемые решения, используя полученные модели.

Язык UML

Унифицированный язык моделирования (*Unified Modeling Language* — *UML*) — это стандартный инструмент для разработки «чертежей» программного обеспечения. Его можно использовать для визуализации, спецификации, конструирования и документирования артефактов программных систем. *UML* подходит для моделирования любых систем — от информационных систем масштаба предприятия до распределенных *Web*-приложений и даже встроенных систем реального времени.

В нотации языка *UML* определены следующие виды канонических диаграмм:

- 1 вариантов использования (*use case diagram*);
- 2 классов (*class diagram*);
- 3 кооперации (*collaboration diagram*);
- 4 последовательности (*sequence diagram*);
- 5 состояний (*statechart diagram*);
- 6 деятельности (*activity diagram*);
- 7 компонентов (*component diagram*);
- 8 развертывания (*deployment diagram*).

Перечень этих диаграмм и их названия являются каноническими в том смысле, что представляют собой неотъемлемую часть графической нотации языка *UML*. Более того, процесс ООАП неразрывно связан с процессом построения этих диаграмм. При этом совокупность построенных таким образом диаграмм является самодостаточной в том смысле, что в них содержится вся информация, которая необходима для реализации проекта сложной системы. Каждая из этих диаграмм детализирует и конкретизирует различные представления о модели сложной системы в терминах языка *UML*.

Диаграмма вариантов использования

Создание диаграммы вариантов использования имеет следующие цели:

- Определить общие границы и контекст моделируемой предметной области на начальных этапах проектирования системы.
- Сформулировать общие требования к функциональному поведению проектируемой системы.
- Разработать исходную концептуальную модель системы для ее последующей детализации в форме логических и физических моделей.
- Подготовить исходную документацию для взаимодействия разработчиков системы с ее заказчиками и пользователями.

Актер (*actor*) – согласованное множество ролей, которые играют внешние сущности по отношению к вариантам использования при взаимодействии с ними (это может быть любой объект, субъект или система, взаимодействующая с моделируемой бизнес-системой извне, т.е. человек, техническое устройство, программа и т.п.).

Вариант использования – внешняя спецификация последовательности действий, которые система или другая сущность могут выполнять в процессе взаимодействия с актерами (он определяет набор действий, совершаемый системой при диалоге с актером).

На рисунке 13 приведена диаграмма вариантов использования для разрабатываемой системы.

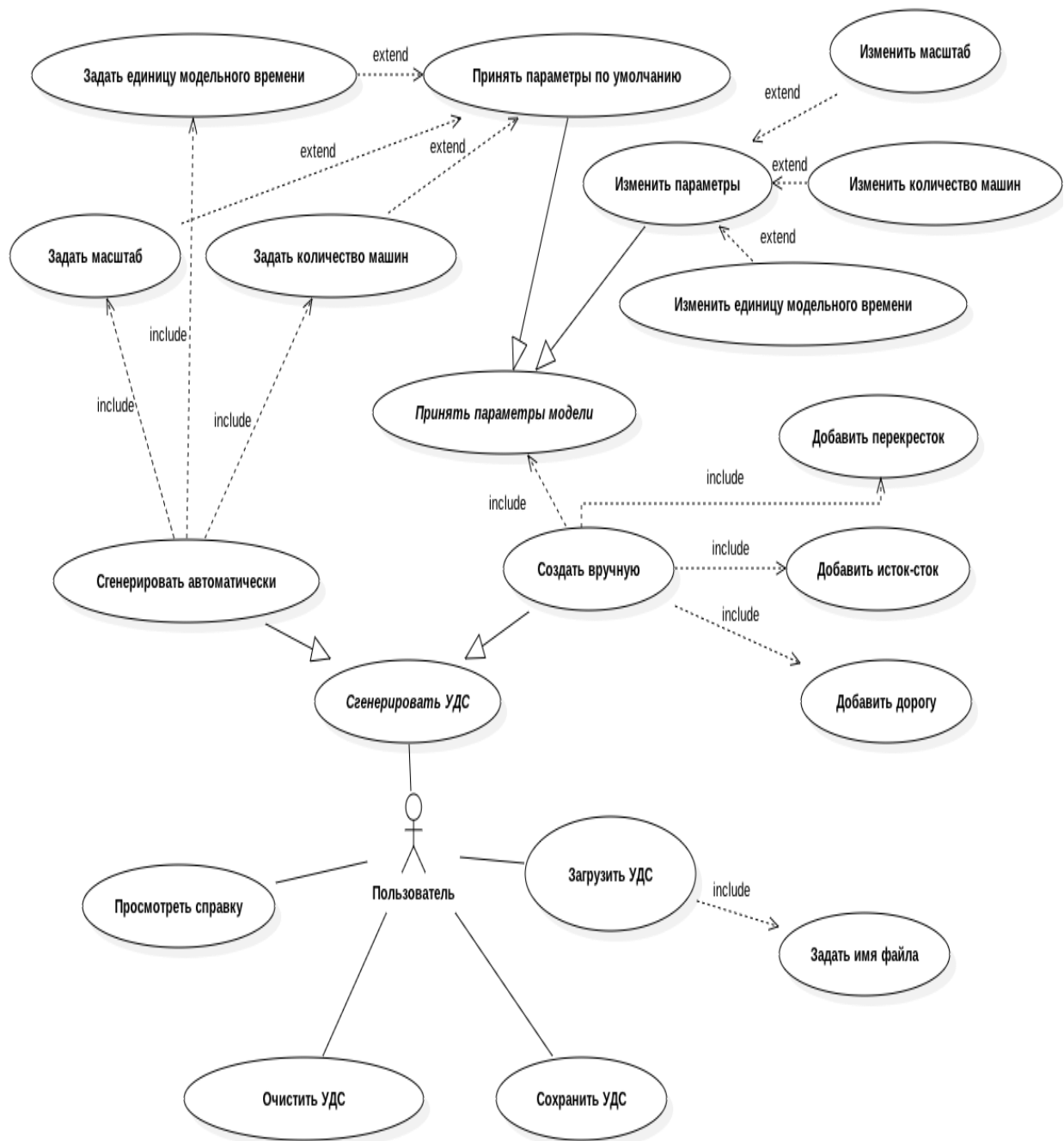


Рисунок 13. Общая диаграмма вариантов использования системы

Цель спецификации варианта использования заключается в том, чтобы зафиксировать некоторый аспект или фрагмент поведения проектируемой системы без указания особенностей реализации данной функциональности. В этом смысле каждый вариант использования соответствует отдельному сервису, который предоставляет моделируемая система по запросу актера, т. е. определяет один из способов применения системы. Сервис, который инициализируется по запросу актера, должен представлять собой законченную последовательность действий. Это означает, что после того как система закончит обработку запроса актера, она должна возвратиться в исходное состояние, в котором снова готова к выполнению следующих запросов.

После входа в программу перед пользователем сразу появляется автоматически сгенерированное УДС *ITSGIS*. Пользователь может кликнуть левой кнопкой мышки по следующим пунктам меню:

- кнопка «Сгенерировать УДС» автоматически создаст новое УДС *ITSGIS*;
- кнопка «Очистить УДС» автоматически очистит поле;
- кнопка «Сохранить УДС» позволит сохранить текущее УДС в файл *json* формата;
- кнопка «Загрузить УДС» позволит загрузить сохраненное ранее УДС *ITSGIS* из файла с форматом *json*.

Система поддерживает ручное создание УДС *ITSGIS*. Для этого необходимо зажать клавишу «*Shift*» и кликнуть по ячейке поля.

Автоматический и ручной режим поддерживает параметры моделирования по умолчанию. Однако в любой момент пользователь сам может изменить их, кликнув левой кнопкой мыши на соответствующие пункты меню.

Диаграмма классов

Диаграммы классов – это тип диаграмм, которые создаются при моделировании объектно-ориентированных систем, они показывают набор классов, интерфейсов и коопераций, а также их связи. На практике диаграммы классов применяют для моделирования статического представления системы, они служат основой для целой группы взаимосвязанных диаграмм – диаграмм компонентов и диаграмм размещения.

Диаграмма классов разрабатываемой системы представлена на рисунке 14.

Разрабатываемая система должна содержать класс *ITSGIS*, в котором хранится текущее состояние системы с параметрами моделирования такими как: время; количество машин.

Также существуют классы, соответствующие предметной области такие как: машина; дорога; перекресток; сток/исток.

Эти классы помогают для реализации состояния системы. В них хранится информация о местоположении объекта класса на *ITSGIS*. Также классы предоставляют методы для реализации функций системы.

Диаграмма состояний

Главное предназначение этой диаграммы – описать возможные последовательности состояний и переходов, которые в совокупности характеризуют поведение элемента модели в течение его жизненного цикла. Диаграмма состояний представляет динамическое поведение сущностей, на основе спецификации их реакции на восприятие некоторых конкретных событий. Системы, которые реагируют на внешние действия от других систем или от поль-

зователей, иногда называют реактивными. Если такие действия инициируются в произвольные случайные моменты времени, то говорят об асинхронном поведении модели. После запуска приложения отображается основное меню программы, система ждет действий пользователя. При нажатии кнопки «Сгенерировать УДС» пользователю отображается сгенерированное *ITSGIS*, при нажатии клавиши «Shift» происходит ручное составление УДС, при изменениях, соответствующих скроллбаров происходит изменение параметров при моделировании, при нажатии кнопки «Справка» отображается справочная информация о системе. По нажатию на «X» форма закрывается.

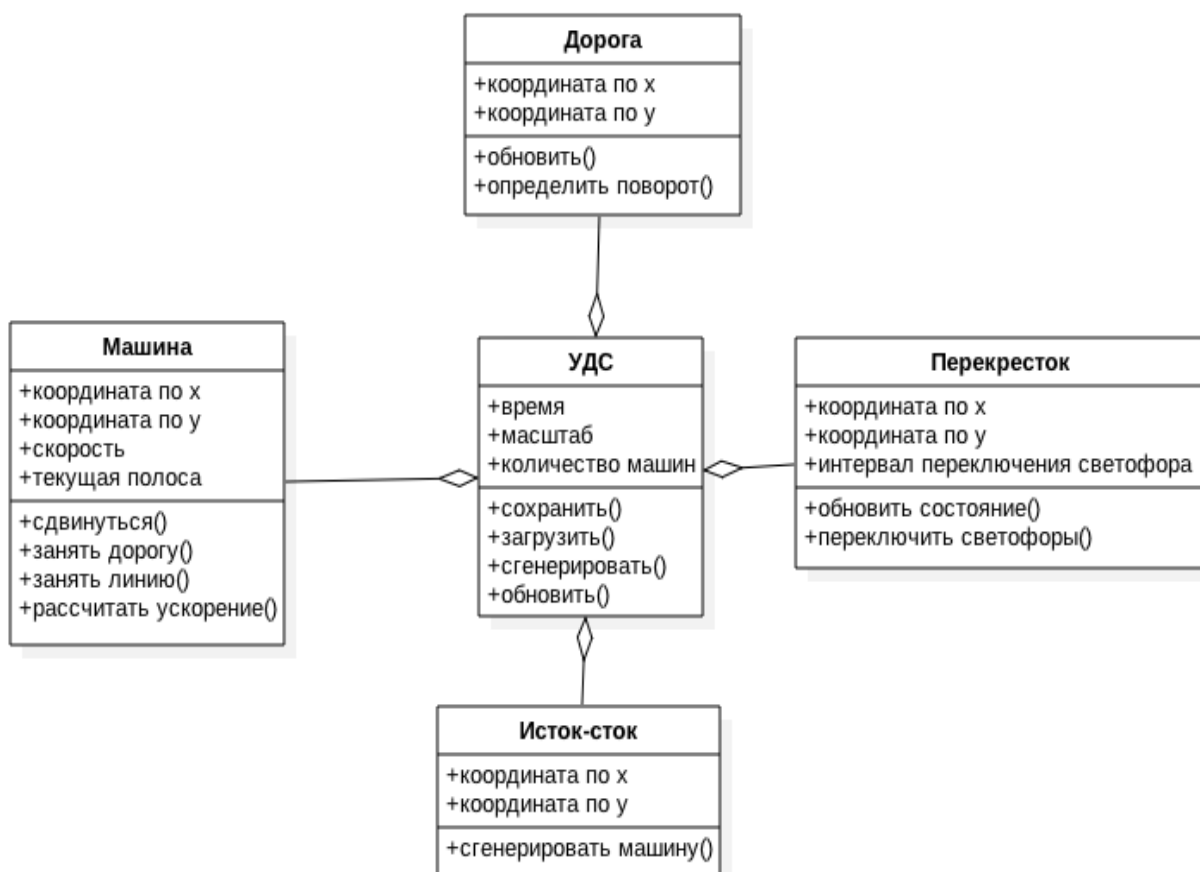


Рисунок 14. Диаграмма классов системы

На рисунках 15, 16 и 17 приведены диаграммы состояний.

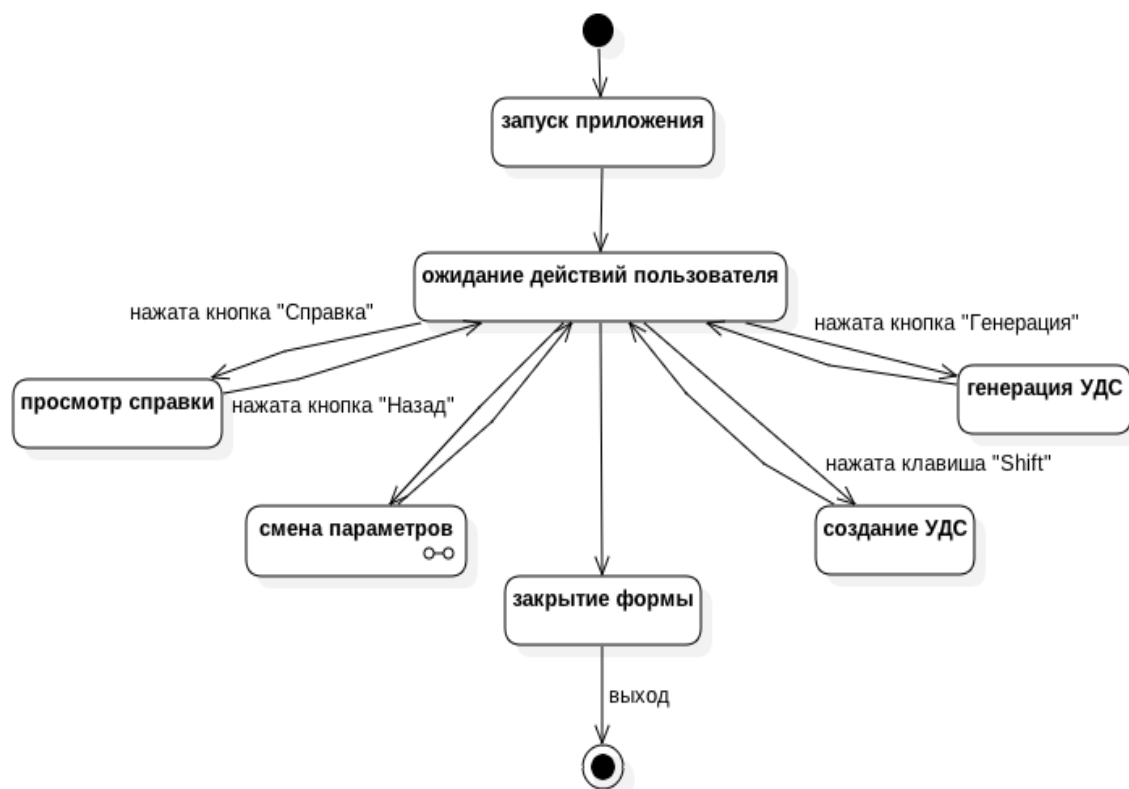


Рисунок 15. Общая диаграмма состояний системы

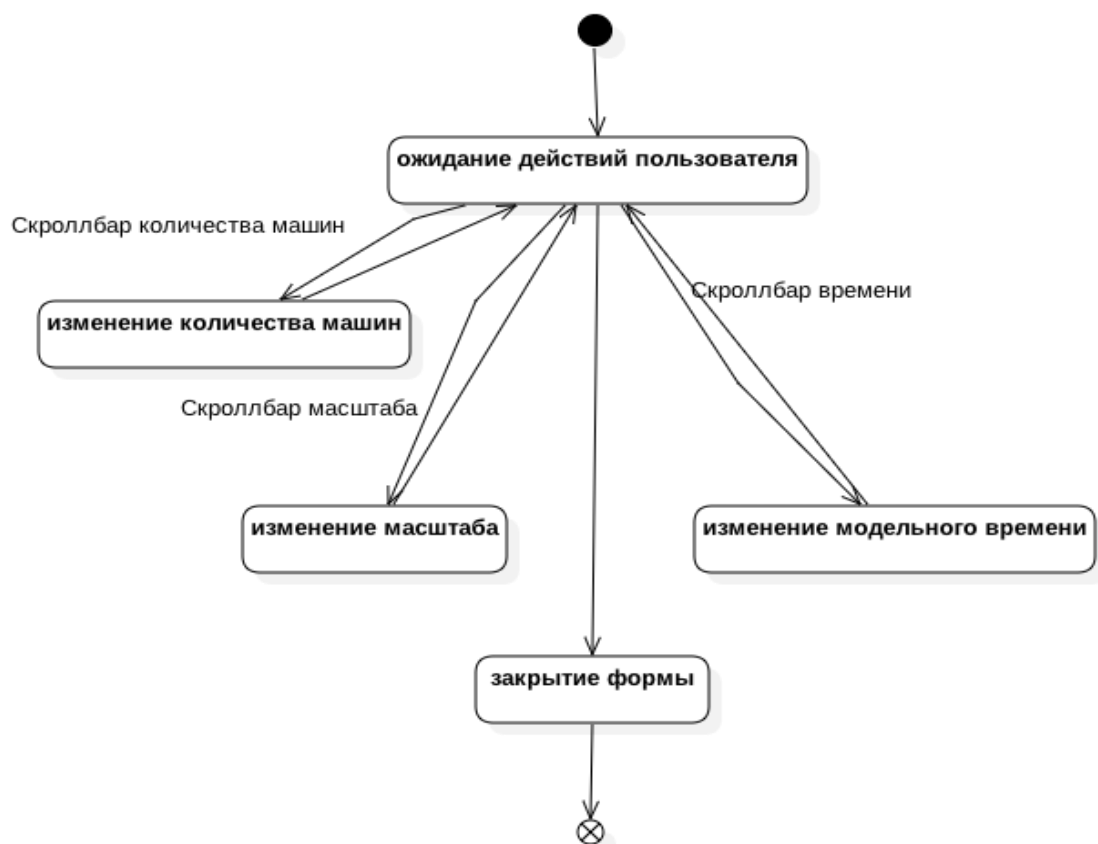


Рисунок 16. Диаграмма состояний «Смена параметров»

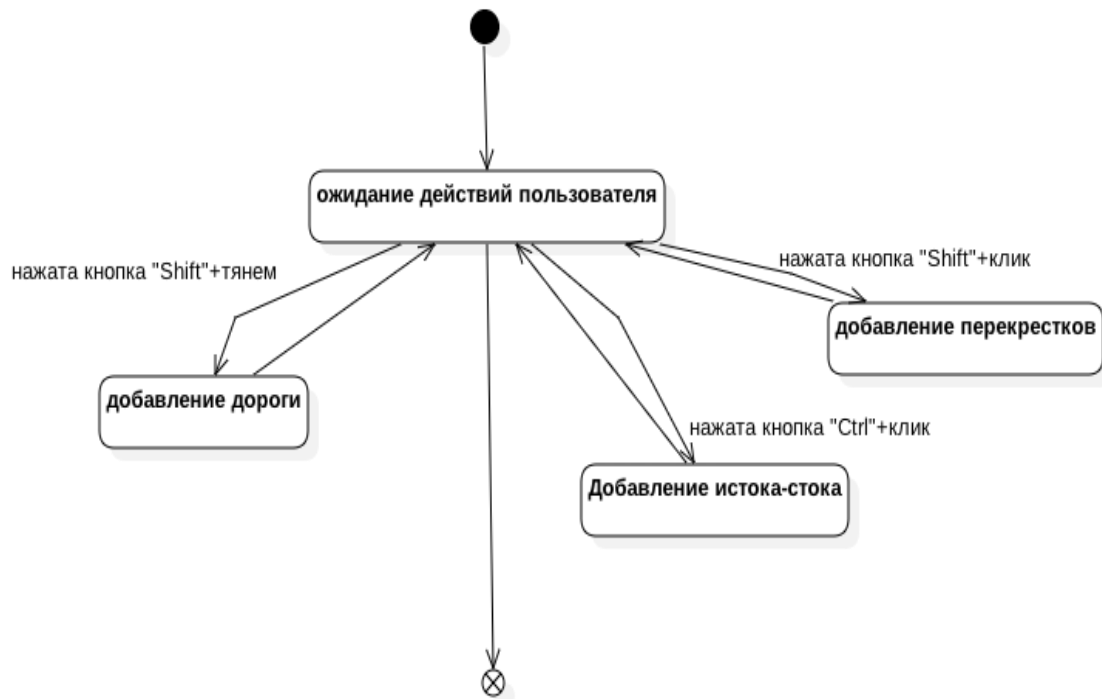


Рисунок 17. Диаграмма состояний «Ручное составление УДС»

Диаграмма деятельности

Диаграммы деятельности – это один из пяти видов диаграмм, применяемых в *UML* для моделирования динамических аспектов поведения системы. Диаграмма деятельности – это, по существу, блок-схема, которая показывает, как поток управления переходит от одной деятельности к другой. Как правило, они применяются, чтобы промоделировать последовательные (а иногда и параллельные) шаги вычислительного процесса. С помощью диаграмм деятельности можно также моделировать жизнь объекта, когда он переходит из одного состояния в другое в разных точках потока управления. Диаграммы деятельности могут использоваться самостоятельно для визуализации, специфицирования, конструирования и документирования динамики совокупности объектов, но они пригодны также и для моделирования потока управления при выполнении некоторой операции.

Состояние деятельности – состояние в графе деятельности, которое служит для представления процедурной последовательности действий, требующих определенного времени.

Диаграмма деятельности системы представлена на рисунке 18.

Можно построить несколько диаграмм деятельности для одной и той же системы, причем каждая из них будет фокусироваться на разных аспектах системы, показывать различные действия, выполняющиеся внутри ее. Читатель, конечно же, понял, что, когда мы говорим о динамике, мы подразумеваем поведение системы в целом или ее частей. Говоря более формально, диаграммы

активности, в общем-то, не имеют монополии на описание поведенческих особенностей динамических частей системы.

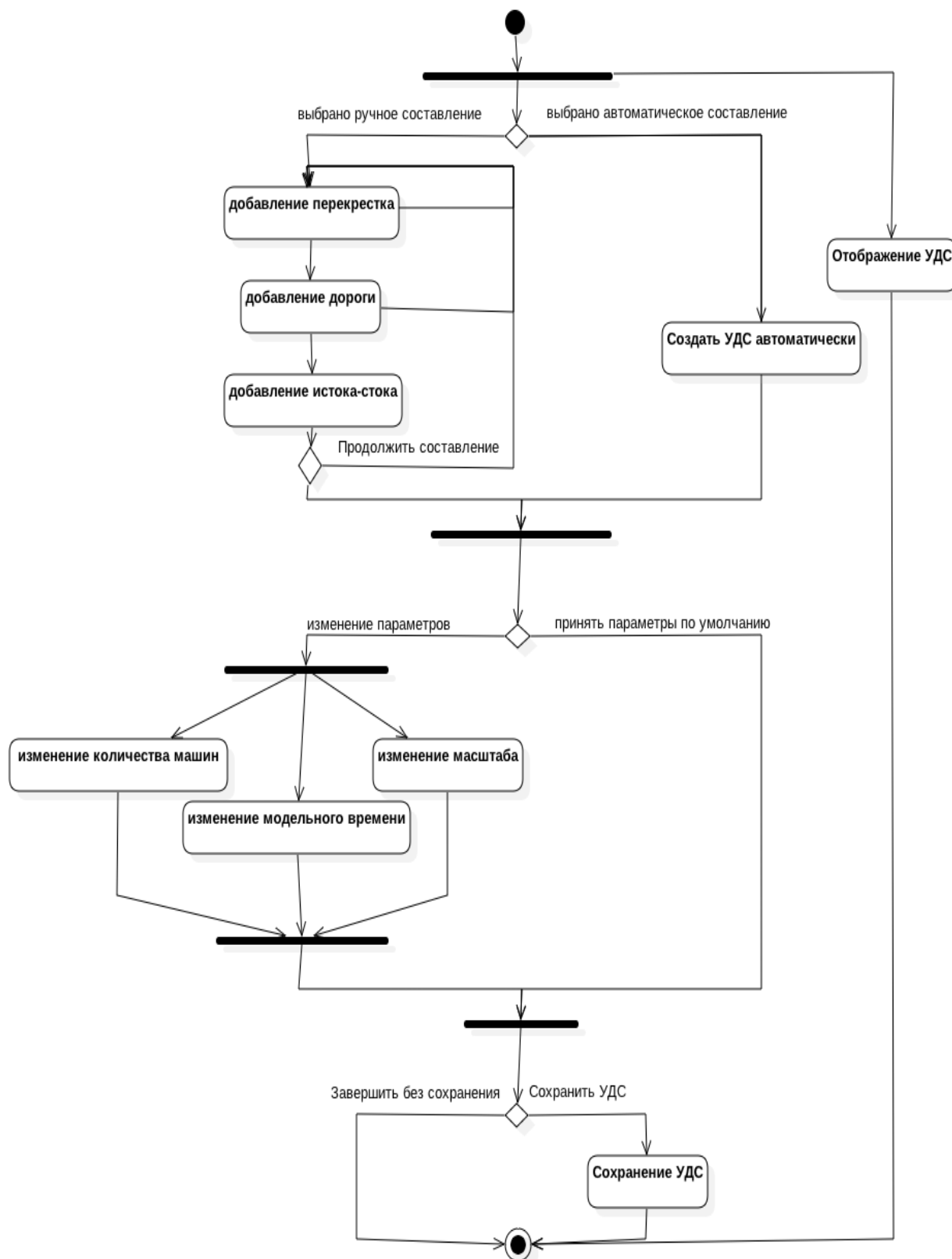


Рисунок 18. Диаграмма деятельности системы

Для этой же цели могут использоваться еще диаграммы прецедентов, последовательности, кооперации и состояний. Создание УДС *ITSGIS* состоит

из следующих этапов: выбор ручного (с самостоятельным построением) либо автоматического режима составления УДС. После составления *ITSGIS* можем изменить параметры моделирования, которые были даны нам по умолчанию. Результаты всех действий будут немедленно отображены на экране. После чего можно воспользоваться функцией сохранения УДС или загрузки готовой УДС из файла.

Диаграмма последовательности

Диаграмма последовательности – это диаграмма, чаще всего, описывающая один сценарий приложения. На диаграмме изображаются экземпляры объектов и сообщения, которыми они обмениваются в рамках одного прецедента (*use case*).

На диаграмме последовательности, каждый участник представлен вместе со своей линией жизни (*lifecycle*), это вертикальная линия под объектом, вертикально упорядочивающая сообщения на странице. Все сообщения на диаграмме следует читать сверху вниз. Каждая линия жизни имеет полосу активности (прямоугольники), которая показывает интервал активности каждого участника при взаимодействии.

Сообщения можно разделить на 2 вида: синхронные (*synchronous message*) – требующие возврата ответа и асинхронные (*asynchronous message*) – ответ не требуется и вызывающий объект может продолжать работу. На диаграмме синхронные вызовы обозначаются закрашенными стрелочками. Асинхронные – не закрашенными или половинными стрелочками.

У первого сообщения, в нашей диаграмме, нет участника, пославшего его, поскольку оно приходит от неизвестного источника. Такое сообщение называется найденным сообщением (*found message*).

Обратной пунктирной стрелкой показан возврат ответа на сообщение *message* (т.к. сообщение *message* является синхронным). Лучше применять изображение возврата только в тех случаях, когда это поможет лучше понять устройство взаимодействия. Во всех остальных случаях, стоит опускать изображения возвратов, т.к. они будут вносить некоторую неразбериху. Просто, при использовании синхронного сообщения, стоит помнить, что у него всегда есть возврат.

Если в сообщении требуется передать параметры, то они указываются в скобках через запятую, с указанием типа параметра. На рисунках 19, 20 приведены диаграммы последовательностей сценариев программы *ITSGIS*.

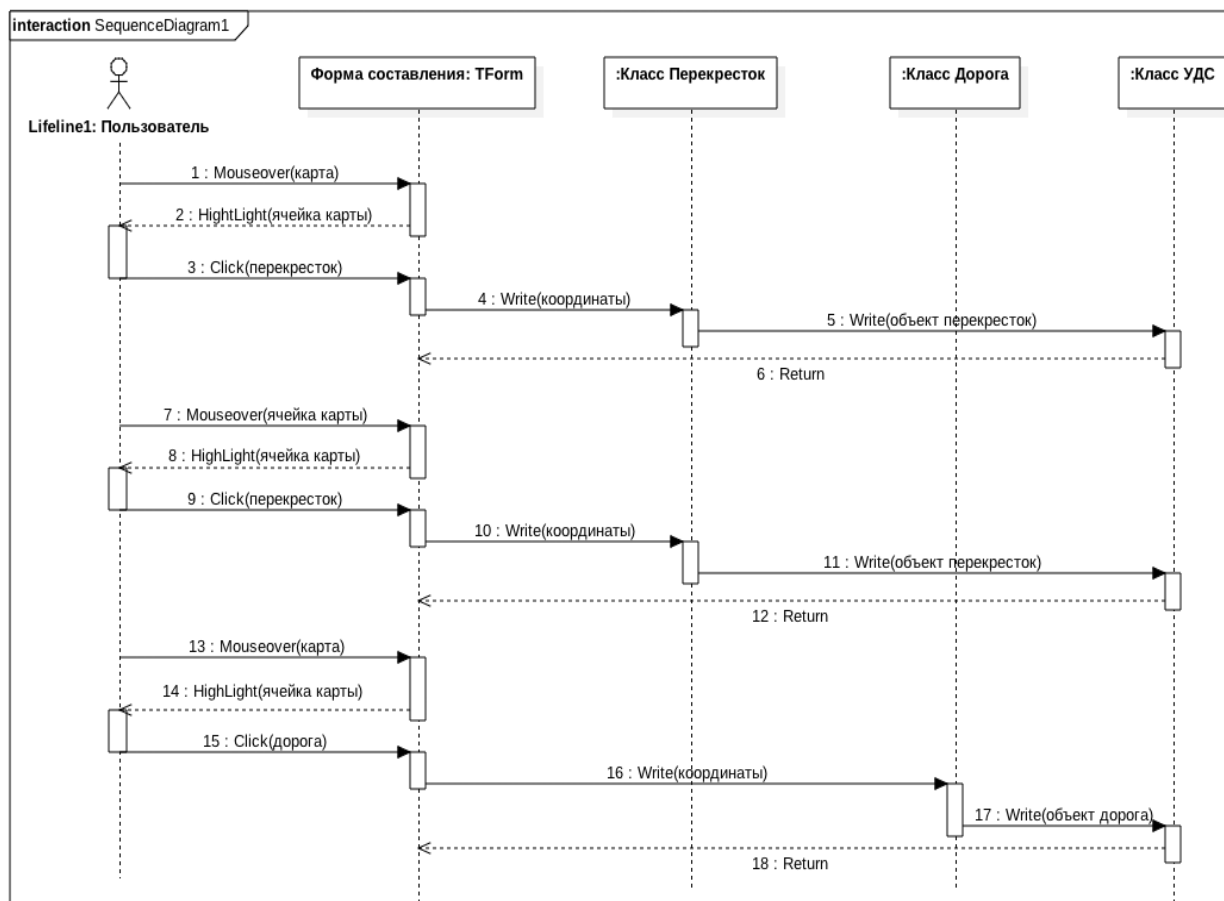


Рисунок 19. Диаграмма последовательности
для варианта использования «Ручное составление УДС»

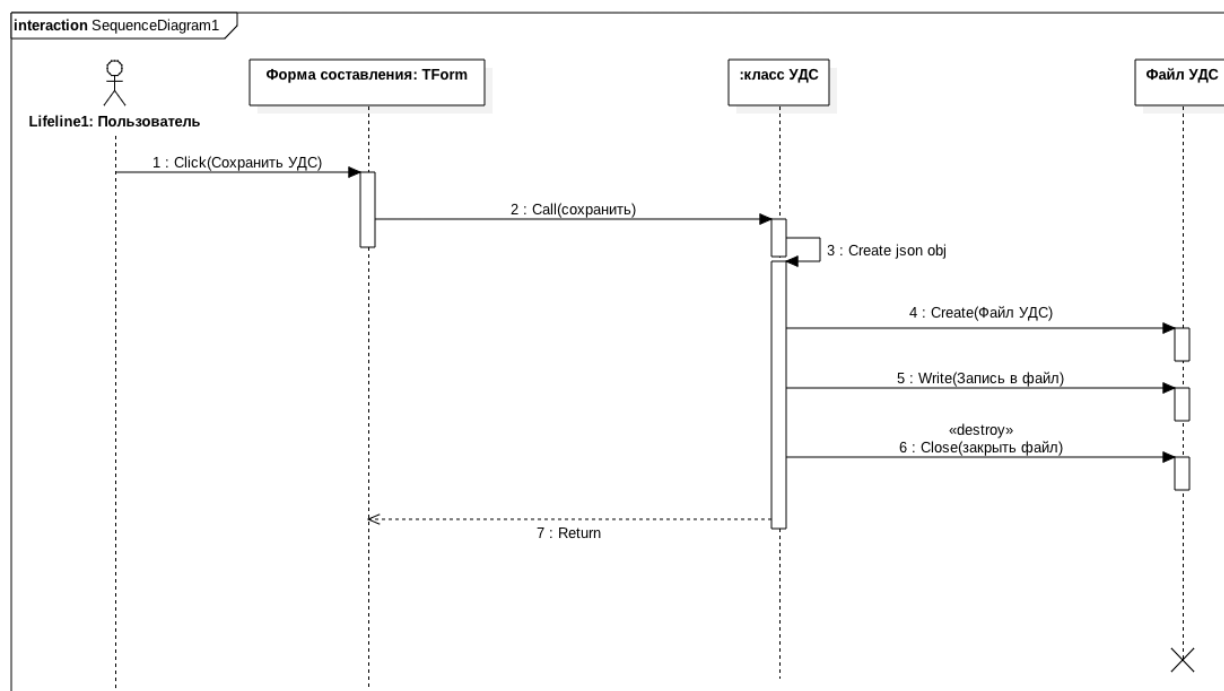


Рисунок 20. Диаграмма последовательности
для варианта использования «Сохранить УДС»

Выбор и обоснование комплекса программных средств

Скриптовый язык *JavaScript* создан компанией *Netscape* в 1995 году. Его первое название было «*LiveScript*» (переводится как «живой сценарий»), но, так как уже в то время, язык программирования *Java* набирал популярность, то разработчики решили для удобства использования скриптов, сделать основные конструкции похожими, соответственно, сам язык переименовали в *JavaScript*. Если вас интересуют уроки по *javascript*, то советую сайт sitear.ru. Там находятся отличные материалы по этой теме.

Первое, что дает нам *JavaScript*, это возможность значительно снизить нагрузку на веб-сервер за счёт того, что основные сценарии для работы со страницей будут выполняться на стороне пользователя. Такие скрипты применяют для создания интерактивных страниц, для написания программ, например, легкой браузерной игры или калькулятора. Большинство сервисов и сайтов применяют *JavaScript* для определения браузера, операционной системы и так далее, для того, чтобы более корректно отобразить страницу на конкретном устройстве. Особенно это актуально для обозревателя *Internet Explorer*. Именно *JavaScript* позволяет взаимодействовать с *cookie*, хранить часть информации на стороне пользователя, и использовать ее при повторном посещении им сайта.

Еще один способ применения: проверка форм ввода информации на правильность без перезагрузки страницы. Теперь кратко о плюсах:

- В отличие от *HTML5*, поддерживается практически всеми браузерами.
- Прост в освоении и применении.
- Огромная библиотека готовых полезных скриптов.
- Позволяет разгрузить сервер, ведь выполняется на стороне пользователя.

WebStorm – интегрированная среда разработки на *JavaScript*, *CSS* & *HTML* от компании *JetBrains*, разработанная на основе платформы *IntelliJ IDEA*. *WebStorm* обеспечивает автодополнение, анализ кода на лету, навигацию по коду, рефакторинг, отладку, и интеграцию с системами управления версиями. Важным преимуществом интегрированной среды разработки *WebStorm* является работа с проектами (в том числе, рефакторинг кода *JavaScript*, находящегося в разных файлах и папках проекта, а также вложенного в *HTML*). Поддерживается множественная вложенность (когда в документ на *HTML* вложен скрипт на *Javascript*, в который вложен другой код *HTML*, внутри которого вложен *Javascript*) – то есть в таких конструкциях поддерживается корректный рефакторинг.

Выбор операционной системы

Были рассмотрены две наиболее популярные среди пользователей операционные системы: *Windows* и *Mac OS*.

Mac OS – это операционная система, разработанная компанией *Apple*, и в настоящее время является второй наиболее часто используемой *ОС* после *Windows*. У нее менее 20% доли рынка. *Mac OS* в отличие от *Windows* основана на *Unix*. Таким образом, *Mac OS* считается частью семейства *Unix ОС*, как и *Linux*. К основным плюсам можно отнести скорость, надежность и стабильность работы, отсутствие вирусов, удобность интерфейса, предварительный (быстрый) просмотр и большое количество полезных программ.

Windows – это одна из самых известных операционных систем, разработанных *Microsoft*. В 9 из 10 домах и предприятий в настоящее время по крайней мере один компьютер с *Windows*. *ОС Windows* первоначально был основан на *MS-DOS*. Все последующие *ОС Windows* основаны на *Windows NT*. К основным плюсам относится большое количество дополнительных программ, множество игр и привычность, простота работы.

Реализация системы

Интерфейс – система правил и средств, регламентирующая и обеспечивающая взаимодействие нескольких процессов или объектов.

Пользовательский интерфейс (ПИ) – система правил и средств, регламентирующая и обеспечивающая взаимодействие программы с пользователем. В понятие пользовательского интерфейса (ПИ) входит не только, и даже не столько, картинка на экране – трехмерная, анимированная или просто выполненная в модном дизайне, а способы взаимодействия пользователя с системой. В целях предоставления пользователю удобного доступа к функциональности системы авторами был разработан интерфейс, максимально удобный для работы пользователя с системой.

Разработка и описание пользовательского меню

Меню состоит из 5 кнопок:

- сохранить;
- загрузить;
- очистить;
- сгенерировать;
- справка.

Также для изменения параметров моделирования, используется 3 скролбара:

- количество;
- масштаб;
- модельное.

При нажатии на кнопку «X» основная форма закрывается. Программа автоматически выводит всплывающее меню, предлагающее пользователю генерацию УДС либо ручное составление.

На рисунке 21 представлена главная форма приложения.

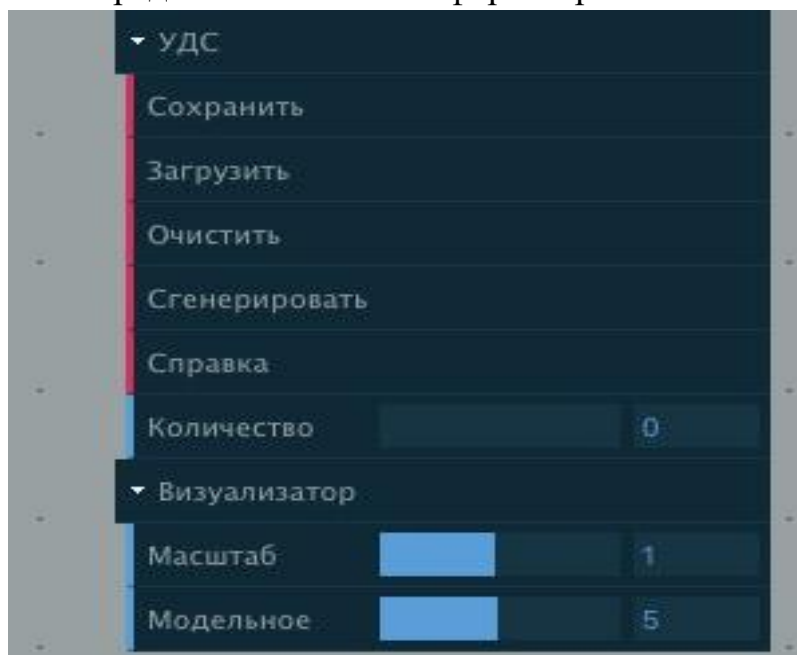


Рисунок 21. Главная форма приложения

Пользователь может изменять параметры моделирования прямо в процессе работы программы. Для этого достаточно изменить положение скроллбаров с значений, установленных по умолчанию, на нужного пользователя (Рисунок 22).

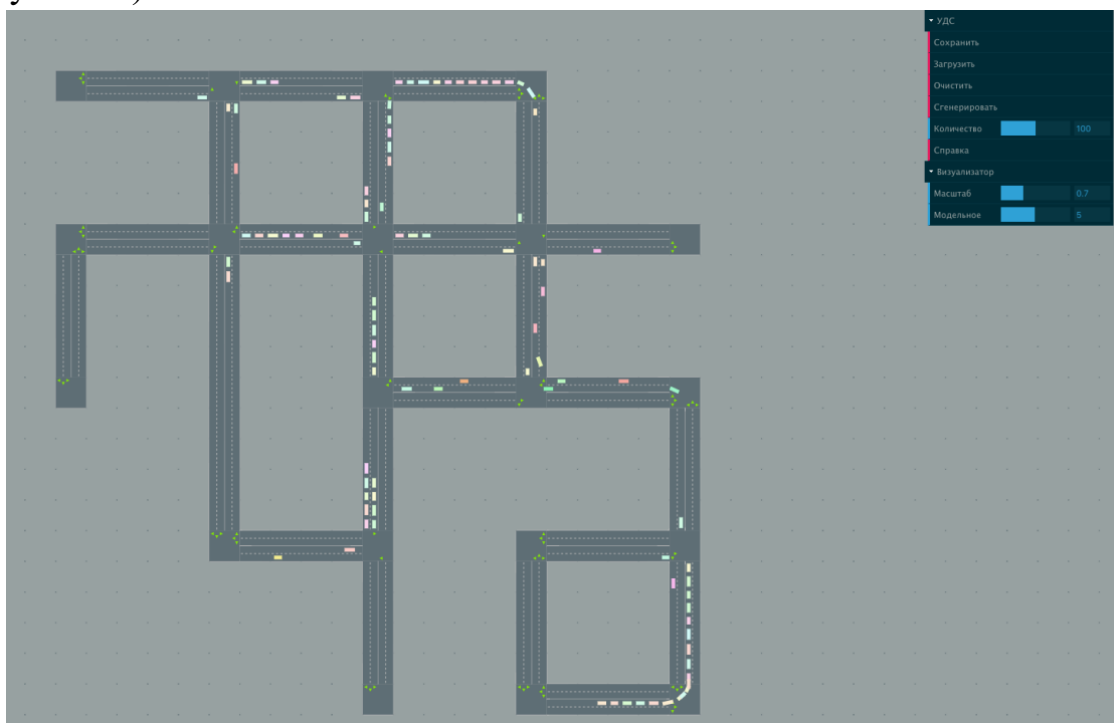


Рисунок 22. Форма сгенерированной *ITSGIS*

При нажатии на кнопку «Справка» открывается справочное окно, содержащее информацию об авторах и документацию. Данное окно изображено на

рисунке 23. Вернуться в главное меню можно по нажатию кнопки «Главное меню».

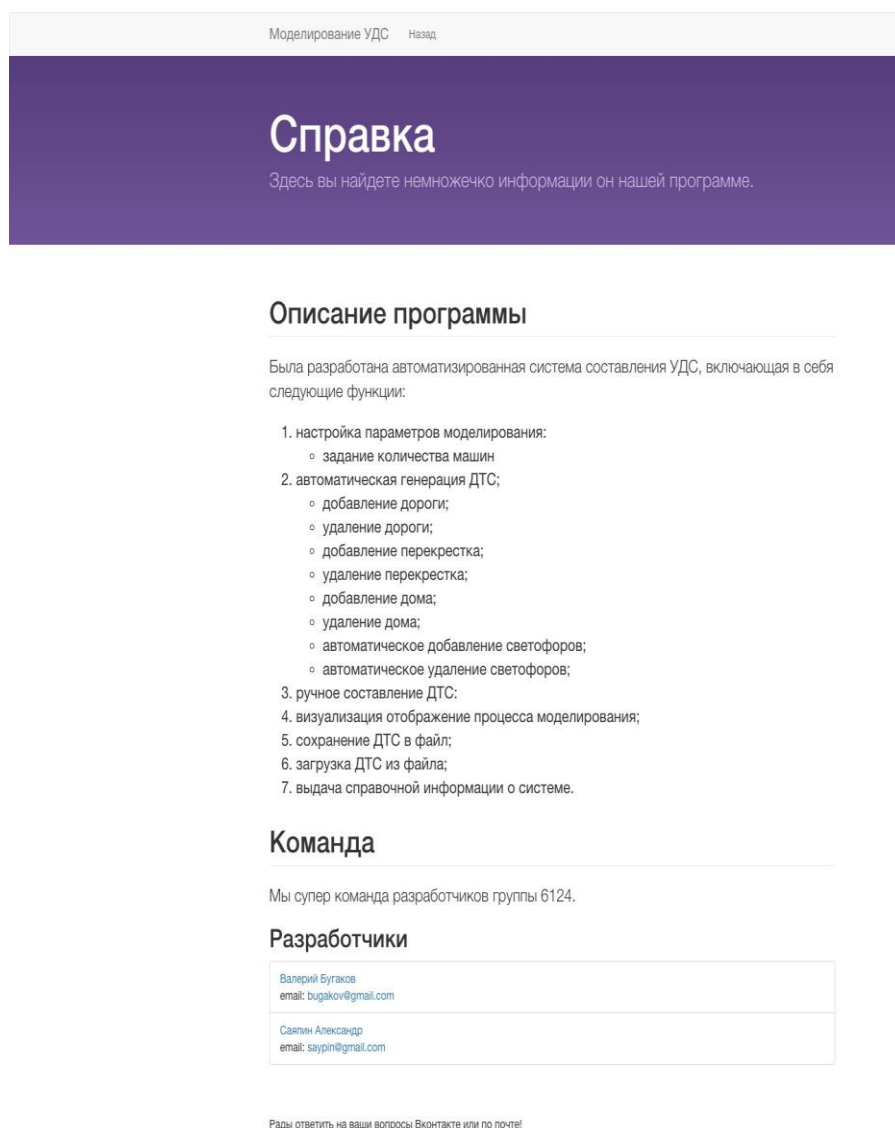


Рисунок 23. Форма справки

Описание контрольного примера

Рассмотрим в качестве контрольного примера ручное составление УДС.

Для создания новой *ITSGIS* УДС нам необходимо создать достаточное количество перекрестков и дорог. Для этого нужно в ручном режиме нажать клавишу «Shift» после чего кликнуть по ячейки поля. Система создаст и автоматически отобразит перекресток. Результат этого можно увидеть на рисунке 24. Повторяя последовательно это действие, мы можем создать дополнительные объекты перекрестков, после чего мы можем соединить дорогу. Для этого нам надо нажать клавишу «Shift» и протянуть мышкой от одного перекрестка к другому. Система создаст и автоматически отобразит дорогу соединяющая перекрестки. Выполняя последовательность этих действий, мы можем создать свою собственную УДС. После построения можем сохранить

построенную *ITSGIS* путем нажатия кнопки «Сохранить». После чего сохраненная УДС будет доступна для загрузки пользователю после нажатия кнопки «Загрузить». Завершить работу с приложением можно, нажав кнопку «X».

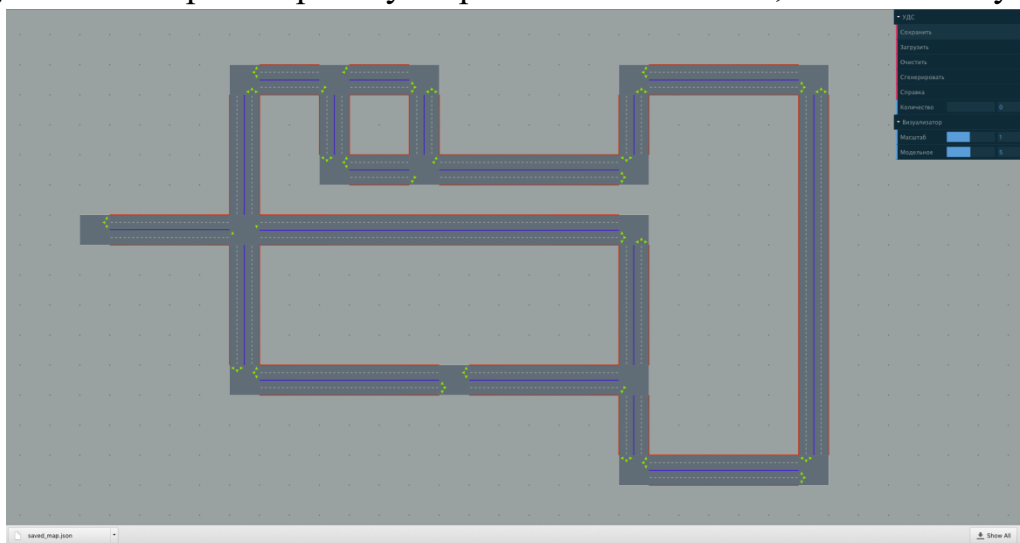


Рисунок 24. Сохранение *ITSGIS* в файл

Выбор и обоснование комплекса технических средств

Расчет объема занимаемой памяти включает в себя расчет необходимого объема оперативной памяти, а также памяти жесткого диска. Необходимый объем внешней памяти рассчитывается по следующей формуле:

$$V_{ЖД} = V_{ОС} + V_{ПР} + V_{СПО},$$

где $V_{ОС}$ – размер системных файлов операционной системы;

$V_{ПР}$ – объем памяти, занимаемый файлами приложения;

$V_{СПО}$ – объем памяти, занимаемый всем необходимым сопутствующим программным обеспечением.

Операционная система *Windows 7* требует $V_{ОС}$ 9,43 Гб памяти. Файлы приложения занимают $V_{ПР} \sim 2,73$ Мб. Объем памяти, занимаемый всем необходимым сопутствующим программным обеспечением равен $V_{СПО} = 250$ Мб.

Исходя из этих данных, рассчитаем требуемый объем памяти жесткого диска:

$$V_{ЖД} = 9,43 \text{ Гб} + 2,73 \text{ Мб} + 250 \text{ Мб} = 9,683 \text{ Гб}.$$

Для расчета необходимого объема ОЗУ используется следующая формула:

$$V_{ОЗУ} = V_{ОС} + V_{ПР}.$$

Операционная система занимает 1024 Мб, приложение – 401 Мб. Исходя из этих значений, рассчитаем требуемый объем оперативной памяти:

$$V_{ОЗУ} = 1024 \text{ Мб} + 401 \text{ Мб} = 1425 \text{ Мб}.$$

Минимальные требования, предъявляемые к системе

Для функционирования системы необходимо:

- 1) персональный компьютер с ОС *Windows XP* или выше, либо *Mac OS*;

- 2) не менее 512 Мб оперативной памяти;
- 3) 9,683 Гб на жестком диске;
- 4) дисплей с разрешением не менее 1024 x 768 пикселей;
- 5) клавиатура или иное устройство ввода;
- 6) мышь или иное манипулирующее устройство.

Диаграмма компонентов

Диаграмма *ITSGIS* описывает особенности физической реализации системы в момент перехода от логического представления к конкретной реализации системы. Диаграмма компонентов позволяет определить архитектуру разрабатываемой системы, установив зависимости между программными компонентами, в роли которых может выступать исходный, бинарный и исполняемый код. Основными графическими элементами диаграммы компонентов являются компоненты, интерфейсы и зависимости между ними. На рисунке 25 приведена диаграмма компонентов разрабатываемой системы.

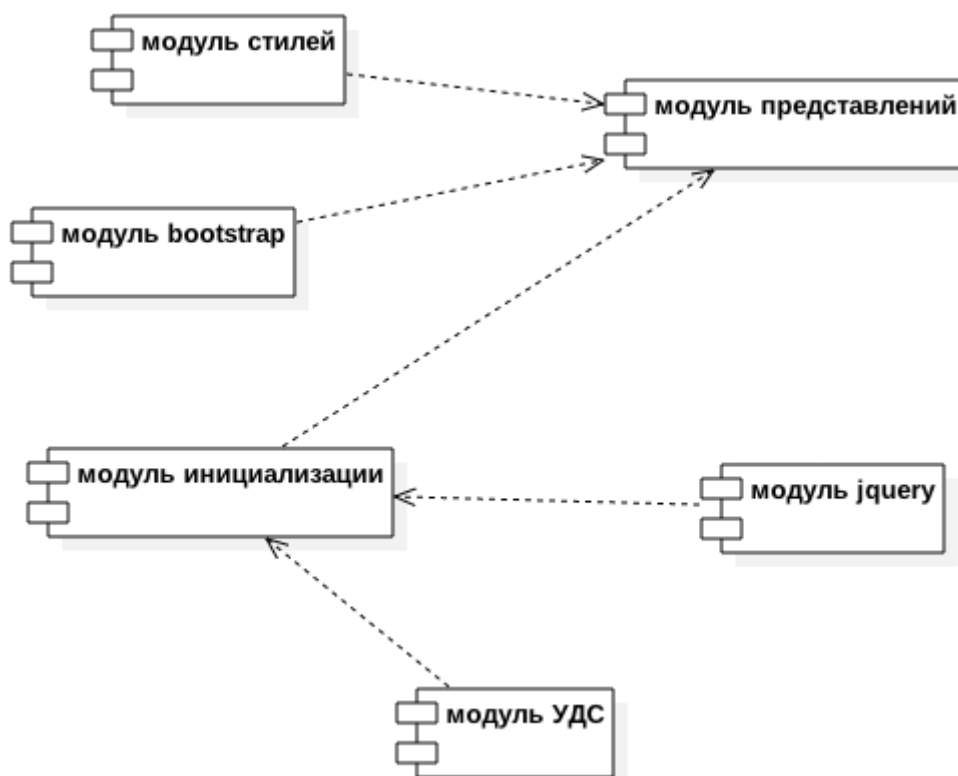


Рисунок 25. Диаграмма компонентов

В программе представлены 6 связанных компонентов:

- модуль стилей, в котором хранятся все стили, используемые приложением;
- модуль *bootstrap* – настраиваемый *HTML*, *CSS*, и *Javascript* инструментарий для корректного отображения данных;
- модуль представления для манипулирования внешним видом сайта, а именно его контентом с различного типа данными;

- модуль инициализации, в котором происходит инициализация переменных;
- модуль *jquery*, включающий в себя множество полезных модулей, таких как модуль подсветки активного поля формы, модуль автоматического ресайза картинок и другие;
- модуль *ITSGIS*, в котором происходит работа с УДС.

Диаграмма развертывания

Диаграмма развертывания предназначена для визуализации элементов и компонентов программы, существующих лишь на этапе ее исполнения (*runtime*). При этом представляются только компоненты-экземпляры программы, являющиеся исполнимыми файлами или динамическими библиотеками. Компоненты, которые не используются на этапе исполнения, на диаграмме развертывания не показываются.

Диаграмма развертывания содержит графические изображения процессоров, устройств, процессов и связей между ними. В отличие от диаграмм логического представления, диаграмма развертывания является единой для системы в целом, поскольку должна всецело отражать особенности ее реализации. Эта диаграмма, по сути, завершает процесс ООАП для конкретной программной системы и ее разработка, как правило, является последним этапом спецификации модели. Диаграмма развертывания разрабатываемой системы приведена на рисунке 26. В качестве операционной системы может выбрана *Windows 7/8* либо *Mac OS*.

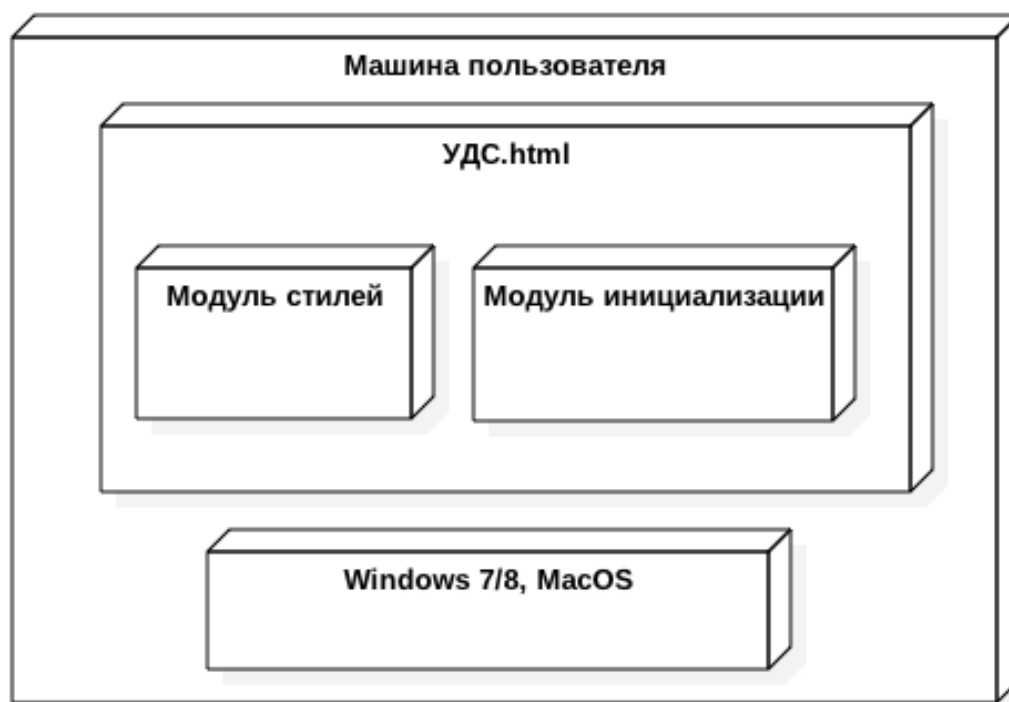


Рисунок 26. Диаграмма развертывания

Заключение

В соответствии с техническим заданием произведен анализ предметной области, реализовано программное и информационное обеспечение, подготовлены контрольные примеры и произведены тестирование и отладка автоматизированной системы микромоделирования движения транспортных средств по *ITSGIS*.

В первом разделе проведены описание и анализ предметной области, обзор существующих систем аналогов, используемых для создания автоматизированных систем моделирования *ITSGIS*. В конце раздела представлена постановка задачи – перечислены функций, которые должны быть реализованы в системе.

Во втором разделе описаны этапы проектирования системы. Разработана структурная схема автоматизированной системы моделирования *ITSGIS*, проведено прототипирование интерфейса пользователя системы и построены соответствующие проекту диаграммы. Также произведен выбор и обоснование комплекса программных средств.

В третьей части происходит реализация системы: разработка и описание интерфейса пользователя, описание контрольного примера. В данном разделе показано как работает автоматизированная система моделирования *ITSGIS*.

Спроектированная система имеет удобный пользовательский интерфейс, легка и надежна в использовании, не требует серьезных аппаратных затрат. При выполнении работы использовалась методология объектно-ориентированного анализа и проектирования, кроме того применялась технология быстрой разработки приложений *RAD* и методология *UML*.

Список использованных источников

1. Зеленко, Л.С. Методические указания к лабораторному практикуму по дисциплине «Программная инженерия» / Л.С. Зеленко. – Самара: СГАУ, 2012. – 51 с.
2. Михеева, Т.И. Интеллектуальная транспортная геоинформационная система *ITSGIS*. Ядро / Т.И. Михеева, С.В. Михеев, О.К. Головнин, А.В. Сидоров, Е.А. Савинов. – Самара : Интелтранс, 2016. – Т.1. – 171 с.
3. Чемодуров, А.С. Обзор средств фильтрации трафика в корпоративной сети / А.С. Чемодуров, А.Ю. Карпутина // М. : Концепт – 2015. № 2. – С. 1-6.
4. Mikheeva, T. Recognition of Urban Transport Infrastructure Objects Via Hyperspectral Images / O. Saprykin, A. Fedoseev, T. Mikheeva // In Proceedings of the 2nd International Conference on Vehicle Technology and

- Intelligent Transport Systems (VEHITS-2016). Rome, Italy: SCITE-PRESS, 2016. – P. 203-208.
5. Михеев, С.В. Интеллектуальная транспортная геоинформационная система ITSGIS / Т.И. Михеева, С.В. Михеев, О.К. Головнин // Современные проблемы безопасности жизнедеятельности: интеллектуальные транспортные системы : материалы IV Международной научно-практической конференции (Казань, 25–26 февраля 2016 г.). – Казань : ГБУ «Научный центр безопасности жизнедеятельности», 2016. – С. 362–368. – ISBN 978-5-85247-837-5.
 6. Кадыров, М.М. Разработка программного модуля для фильтрации сетевого трафика / М.М. Кадыров // М. : Молодой ученый. – 2017. – № 4 (138). – С. 15-18. – URL: <https://moluch.ru/archive/138/38793/>.
 7. Воронина, Е.Е. Фотовидеофиксация как средство профилактики дорожно-транспортных происшествий / Е.Е. Воронина // Вестник НЦБЖД. – 2019. – № 2(40). – С. 73-79.
 8. Гамма, Э. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влассидес // СПб.: Питер, 2001. – 368 с.
 9. Михеева, Т.И. Метод синтеза системы зонального сетевидного управления транспортными процессами / Т.И. Михеева, С.В. Михеев, О.К. Головнин // Известия Самарского научного центра Российской академии наук. – 2016. – Т. 18, № 4 (4). – С. 799-807.
 10. Михеева, Т.И. Системный анализ объектов транспортной инфраструктуры в геоинформационной среде / Т.И. Михеева // Программные продукты и системы. – 2018. – № 1. – С. 12-18.
 11. Головнин, О.К. Применение паттерна «Рабочий поток» в геоинформационной составляющей интеллектуальной транспортной системы / О.К. Головнин // Математика. Компьютер. Образование : тезисы докладов двадцатой международной конференции. – Пущино, 2013. – Режим доступа : <http://www.mce.su/rus/archive/abstracts/mce20/sect101361/doc174037>.
 12. Интерфейс программ. – <http://www.softandweb.ru/index.php/database-development/database-interface.html> (дата обращения 20.11.2024 г.).
 13. Михеева, Т.И. Структурно-параметрический синтез интеллектуальных транспортных систем / Т.И. Михеева. – Самара : Самар. науч. центр РАН, 2008. – 380 с.
 14. Михеев, С.В. Модели наследования в системе управления дорожным движением / Т.И. Михеева, С.В. Михеев // Информационные технологии. – 2001. – № 7. – С. 50-54.
 15. Минниханов, Р.Н. Применение данных ИТС систем для моделирования эффективной транспортной среды в Республике Татарстан / Р.Н.

- Минниханов, С.С. Степанова, В.А. Решетов, А.З. Махмутова // Безопасность дорожного движения. – 2021. – № 3. – С. 8-12.
16. Академия Microsoft: Методы и средства инженерии программного обеспечения. – <http://www.intuit.ru/studies/courses/2190/237/lecture/6118> (дата обращения 20.11.2024 г.).
17. Сапрыкин, О.Н. Построение архитектуры аналитического инструментария интеллектуальной транспортной системы на основе паттернов проектирования / Т.И. Михеева, О.Н. Сапрыкин, О.В. Сапрыкина // Вестник Самарского государственного технического университета. Серия: Технические науки. – 2010. – № 2 (27). – С. 27–35.

Mikheeva T.I.

**MODERN METHODS OF MICROMODELING THE MOVEMENT OF
VEHICLES IN "ITSGIS": DEVELOPMENTS AND DESIGNING
COMPLEX SOFTWARE SYSTEMS**

*Samara, Samara University named after academician S.P. Korolev
IntelTrans*

In this article, during the course project, a system has been developed that allows for automatic or manual simulation of the street and road network (UDS) with specified modeling parameters. The scheme of the simulated UDS can be stored in files and then uploaded to the program. The methods and tools that were used in the implementation of the solutions obtained in the ITSGIS system are disclosed.

Keywords: modeling, street and road network, intensity, geographic information system, analog systems, intersection, road.

УДК 004.02

Михеева Т.И., Смолев А.М.

**РАЗРАБОТКА АДРЕСНЫХ ПЛАНОВ ГОРОДОВ РОССИИ
В ИНТЕЛЛЕКТУАЛЬНОЙ ТРАНСПОРТНОЙ
ГЕОИНФОРМАЦИОННОЙ СИСТЕМЫ «ITS GIS»**

*Самара, Самарский университет имени академика С.П. Королёва
ИнтелТранс*

В данной статье приводятся транспортные проблемы современных городов России во время интеллектуальной транспортной геоинформационной системы «ITS GIS». Разработана система, которая позволяет они чрезвычайно широко применяются и имеют серьезное научное основание.

Ключевые слова: проектирование, улично-дорожная сеть, города, интенсивность, интеллектуальная транспортная геоинформационная система.

Введение

Транспортные проблемы современных городов России имеют сложный спектр причин и разнообразные проявления, что приводит к необходимости теоретического осмысления и широкомасштабных практических исследований. Эти проблемы тесно примыкают к сферам градостроительства, землепользования, экологии и др. Одним из наиболее серьезных обстоятельств является то, что для подавляющего большинства пользователей улично-дорожной сети (УДС) последняя не является областью, позволяющей самостоятельно регулировать затраты времени и бюджетов различного уровня (начиная с семейного и заканчивая региональным). Среди очевидных причин данного обстоятельства является сложный характер коллективного влияния участников улично-дорожного движения на совокупные параметры транспортного потока, а также отсутствие в настоящее время учета затрат времени, проведенного в транспортном потоке, на экономическую эффективность региона.

В настоящее время для решения транспортных проблем применяют самые разнообразные методы и средства. Большинство из них имеет долговременный характер и значительную стоимость. В этом смысле анализ и прогнозирование транспортных потоков и процессов также не является слишком дешевыми средствами. Тем не менее, они чрезвычайно широко применяются и имеют серьезное научное основание.

Существующие методы анализа и обработки статистической информации позволяют выявлять закономерности и анализировать процессы в сложных организационно-технических системах, к которым можно отнести и транспортно-дорожный комплекс. Для этих целей создаются информационные системы, используются имитационные модели, методы статистического анализа и прогнозирования. Однако, результаты анализа будут корректными только в том случае, если информация, на основе которой он производится, будет полной, хорошо структурированной и формализованной. В настоящее время трудности формализации процесса движения транспортного потока стали серьезной причиной отставания результатов научных исследований от требований практики.

В работе решается задача построения адресного плана города в интеллектуальной транспортной геоинформационной системе «ITSGIS».

Основные понятия и определения

Геоинформатика (ГИС) – наука, технология и производственная деятельность по научному обоснованию, проектированию, созданию, эксплуатации и использованию географических информационных систем, по разработке геоинформационных технологий, по приложению ГИС для практических и научных целей.

Интеллектуальные транспортные геоинформационные системы (также ИТСГИС: «ITSGIS» – интеллектуальная транспортная система географическая информационная система) – системы, предназначенные для сбора, хранения, анализа и графической визуализации пространственных данных и связанной с ними информации о представленных в ИТСГИС объектах.

ИТСГИС включают в себя систему управления базами данных (СУБД), редакторов растровой и векторной графики и аналитических средств и применяются в картографии, лесоустройстве, геологии, метеорологии, землеустройстве, экологии, и многих других областях.

Интегрированные ИТСГИС (*integrated GIS, ITSGIS*) совмещают функциональные возможности ИТСГИС и систем цифровой обработки изображений (данных дистанционного зондирования) в единой интегрированной среде.

Структура и классификация ИТСГИС

По территориальному охвату ИТСГИС подразделяют на глобальные, субконтинентальные, национальные, зачастую имеющие статус государственных, региональные, субрегиональные, локальные, или местные. В некоторых случаях, такие территориальные ИТСГИС могут быть размещены в открытом доступе в сети Интернет и называются геопорталами.

По предметной области информационного моделирования выделяются городские, недропользовательские, горно-геологические, природоохранные и т.п. Среди них особое наименование, как особо широко распространённые, получили земельные информационные системы.

Также геоинформационные системы могут быть классифицированы по проблемной ориентации – решаемым научным и прикладным задачам. Такими задачами могут быть инвентаризация ресурсов (в том числе кадастр), анализ, оценка, мониторинг, управление и планирование, поддержка принятия решений, геомаркетинг. Кроме того, интегрированные ИТСГИС совмещают функциональные возможности и систем цифровой обработки изображений (данных дистанционного зондирования) в единой интегрированной среде.

Классификация ИТСГИС приведена на рисунке 1.



Рисунок 1. Классификация ИТСГИС по территориальному охвату

Различают также:

- *полимасштабные*, или *масштабно-независимые ИТСГИС (multiscale)*, основанные на множественных, или полимасштабных представлениях пространственных объектов, обеспечивая графическое или картографическое воспроизведение данных на любом из избранных уровней масштабного ряда на основе единственного набора данных с наибольшим пространственным разрешением;
- *пространственно-временные ИТСГИС (spatio-temporal)*, оперирующие пространственно-временными данными.

Геоинформационные системы включают в себя пять ключевых составляющих: аппаратные средства, программное обеспечение, данные, исполнители и методы.

Аппаратные средства – это компьютер, на котором запущена ИТСГИС. В настоящее время ИТСГИС работают на различных типах компьютерных

платформ, от централизованных серверов до отдельных или связанных сетью настольных компьютеров.

Программное обеспечение ИТСГИС содержит функции и инструменты, необходимые для хранения, анализа и визуализации географической (пространственной) информации. Ключевыми компонентами программных продуктов являются:

- инструменты для ввода и оперирования географической информацией система управления базой данных (*DBMS* или *СУБД*);
- инструменты поддержки пространственных запросов, анализа и визуализации (отображения);
- графический пользовательский интерфейс (*GUI* или *ГИП*) для легкого доступа к инструментам и функциям.

Данные – это наиболее важный компонент. Данные о пространственном положении (географические данные) и связанные с ними табличные данные могут собираться и подготавливаться самим пользователем либо приобретаться у поставщиков. В процессе управления пространственными данными географическая информационная система объединяет (а лучше сказать – совмещает) географическую информацию с данными других типов. Например, с конкретным кусочком электронной карты могут быть связаны уже накопленные данные о населении, характере почв, близости опасных объектов и т. д. (в зависимости от задачи, которую придется решать при помощи ИТСГИС). Причем в сложных, распределенных системах сбора и обработки информации часто с объектом на карте связывают не существующие данные, а их источник, что позволяет в реальном времени отслеживать состояние этих объектов. Такой подход применяется, например, для борьбы с чрезвычайными ситуациями вроде лесных пожаров или эпидемий.

Пространственные данные делятся на два типа:

- позиционные (географические), которые описывают местонахождение объекта на земной поверхности;
- непозиционные (атрибутивные), которые описывают сам объект.

Исполнителями именуют людей, которые работают с программными продуктами и разрабатывают планы их использования при решении реальных задач. Может показаться странным, что люди, работающие с программным обеспечением, рассматриваются как составная часть ИТСГИС, однако в этом есть свой смысл. Дело в том, что для эффективной работы географической информационной системы необходимо соблюдение методов, предусмотренных разработчиками, поэтому без подготовленных исполнителей даже самая удачная разработка может утратить всякий смысл.

Пользователями ИТСГИС могут быть как технические специалисты, разрабатывающие и поддерживающие систему, так и обычные сотрудники

(конечные пользователи), которым ИТСГИС помогает решать текущие каждодневные дела и проблемы.

Методы. Успешность и эффективность (в том числе экономическая) применения ИТСГИС во многом зависит от правильно составленного плана и правил работы, которые составляются в соответствии со спецификой задач и работы каждой организации.

Структура ИТСГИС включает четыре обязательные подсистемы:

- подсистема ввода данных, обеспечивающую ввод и/или обработку пространственных данных, полученных с карт, материалов дистанционного зондирования и т.д.;
- подсистема хранения и поиска, позволяющую оперативно получать данные для соответствующего анализа, актуализировать и корректировать их;
- подсистема обработки и анализа, которая дает возможность оценивать параметры, решать расчетно-аналитические задачи;
- подсистема представления (выдачи) данных в различном виде (карты, таблицы, изображения, блок-диаграммы, цифровые модели местности и т.д.).

Геоинформационный проект – наполнение ИТСГИС пространственными данными и сведениями об объектах в привязке к пространственным данным. Проект может быть реализован на какой-либо из тиражируемых ИТСГИС, либо такая система может быть разработана специально для геоинформационного проекта. Типичные этапы геоинформационного проекта:

- предпроектные исследования, включающие изучение функциональных требований, оценку функциональных возможностей геоинформационных систем, технико-экономическое обоснование;
- системное проектирование, включая стадию пилотного проекта, при необходимости – разработку геоинформационных систем или расширение существующих;
- тестирование на небольшом территориальном фрагменте, или тестовом участке, прототипирование, или создание опытного образца, или прототипа (*prototype*);
- внедрение;
- эксплуатация.

Создание карт и географический анализ не являются чем-то абсолютно новым. Однако технология ИТСГИС предоставляет новый, более соответствующий современности, более эффективный, удобный и быстрый подход к анализу проблем и решению задач, стоящих перед человечеством в целом, и конкретной организацией или группой людей, в частности. Она автоматизирует процедуру анализа и прогноза. До начала применения ИТСГИС лишь

немногие обладали искусством обобщения и полноценного анализа географической информации с целью обоснованного принятия оптимальных решений, основанных на современных подходах и средствах.

Представление данных

Данные в ИТСГИС описывают, как правило, реальные объекты, такие как: дороги, здания, водоемы, лесные массивы. Реальные объекты можно разделить на две абстрактные категории: дискретные (дома, территориальные зоны) и непрерывные (рельеф, уровень осадков, среднегодовая температура). Для представления этих двух категорий объектов используются векторные и растровые данные.

Растровые данные хранятся в виде наборов величин, упорядоченных в форме прямоугольной сетки. Ячейки этой сетки называются пикселями. Наиболее распространенным способом получения растровых данных о поверхности Земли является дистанционное зондирование, проводимое при помощи спутников. Хранение растровых данных может осуществляться в графических форматах, например, *TIFF* или *JPEG*.

Векторные данные обычно имеют намного меньший размер, чем растровые. Их легко трансформировать и проводить над ними бинарные операции. Векторные данные позволяют проводить различные типы пространственного анализа, к примеру, поиск кратчайшего пути в дорожной сети. Наиболее распространёнными типами векторных объектов являются точки, полилинии, многоугольники.

Сравнение растрового и векторного изображений приведено на рис. 2.

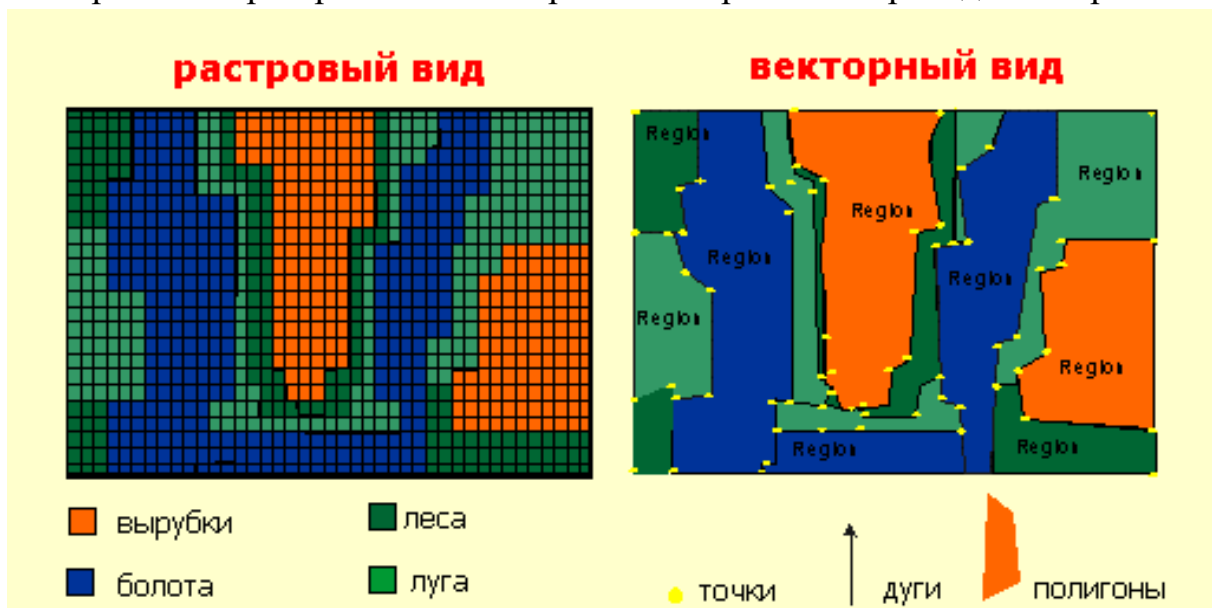


Рисунок 2. Сравнение растрового и векторного изображений

Точки используются для обозначения географических объектов, для которых важно местоположение, а не их форма или размеры. В ИТСГИС то-

чечный объект изображается в виде некоторой геометрической фигуры небольших размеров (квадратик, кружок, крестик), либо пиктограммой, передающей тип реального объекта. Полилинии служат для изображения линейных объектов. Полилиния – ломаная линия, составленная из отрезков прямых. Полилиниями изображаются дороги, железнодорожные пути, реки, улицы, водопровод. Допустимость изображения объектов полилиниями также зависит от масштаба карты. Например, крупная река в масштабах континента вполне может изображаться линейным объектом, тогда как уже в масштабах города требуется её изображение площадным объектом. Характеристикой линейного объекта является длина.

Многоугольники (иногда используется калька «полигоны») служат для обозначения площадных объектов с чёткими границами. Примерами могут служить озера, парки, здания, страны, континенты. Характеризуются площадью и длиной периметра.

Семантические данные могут привязаны к векторным: например, на карте территориального зонирования к площадным объектам, представляющим зоны, может быть привязана характеристика типа зоны. Структуру и типы данных определяет пользователь. На основе численных значений, присвоенных векторным объектам на карте, может строиться тематическая карта, на которой эти значения обозначены цветами в соответствии с цветовой шкалой, либо окружностями разного размера. Непрерывные поля величин могут быть описаны векторными данными. Поля при этом изображаются в виде изолиний или контурных линий. Одним из способов представления рельефа является нерегулярная триангуляционная сетка (*TIN, triangulated irregular networks*). Такая сетка формируется множеством точек с привязанными значениями (в данном случае высота). Значения в произвольной точке внутри сетки получаются путем интерполяции значений в узлах треугольника, в который попадает эта точка.

Описание системы *ITSGIS*

ITSGIS представляет собой многоуровневую, сложноорганизованную интеллектуальную геоинформационную систему, состоящую из множества разнородных систем, взаимодействующих друг с другом.

В основе *ITSGIS* клиент-серверная архитектура, в которой задания или сетевая нагрузка распределены между поставщиками услуг, называемыми серверами, и заказчиками услуг, называемыми клиентами. Клиент и сервер взаимодействуют через компьютерную сеть посредством сетевых протоколов и находятся на разных вычислительных машинах, но могут выполняться также и на одной машине. Программы-сервера ожидают от клиентских про-

грамм запросы и предоставляют им свои ресурсы в виде данных или сервисных функций.

Представление архитектуры клиент-сервер приведено на рисунке 3.



Рисунок 3. Представление архитектуры клиент-сервер

В качестве СУБД (система управления базами данных) в *ITSGIS* используется *PostgreSQL*, поскольку она бесплатно распространяемая, кроссплатформенная, не имеет ограничений по размеру базы данных и поддерживает геоинформационные данные.

В системе *ITSGIS* организовано разграничение прав доступа пользователей на основе георолей: одним пользователям разрешен только просмотр информации, тогда как другие обладают правом модификации данных, причем права определяются как с учетом слоя доступа, так и области на карте.

ITSGIS позволяет разрабатывать дополнительные программные модули (плагины), которые расширяют функциональность системы и дают возможность работать со специализированными геообъектами – точечными, линейными и полигональными геометриями на электронной карте с прикрепленной к ним семантической информацией.

ИТСГИС «ITSGIS». Ядро системы и инструмент просмотра карты с нанесенными объектами, возможностью подключения/отключения слоев, масштабирования, печати электронной карты. Обеспечивает многопользовательскую работу, выполнение геозапросов, разграничение прав доступа, сжатие, шифрацию, выборку геоданных.

Плагин «ITSGIS. Редактор адресного плана». Позволяет редактировать адресный план карты, т.е. указывать привязку геопространственных объектов к их текстовому адресному описанию, редактировать справочники областей, населенных пунктов, районов, улиц, домов, автодорог, перекрестков и др.

Плагин «ITSGIS. Редактор геометрий». Обеспечивает редактирование всех типов геопространственных объектов, их стилей отображения и параметров трансформации (вращение, перемещение и др.).

Плагин «ITSGIS. Учет территорий». Предоставляет возможность редактирования и просмотра информации о закрепленных территориях, хранения фотокопий договоров (контрактов) для закрепленных (очагов аварийно-

сти) на улично-дорожной сети города территорий, а также полнотекстового поиска по договорам и организациям.

Плагин «ITSGIS. Учет ДТП». Предназначен для автоматизации ведения учета дорожно-транспортных происшествий (ДТП) и выявления мест их концентрации

Плагин «ITSGIS. Учет интенсивности». Предназначен для хранения, обработки и визуализации на электронной карте в среде геоинформационной системы «ITSGIS» информации об интенсивности транспортных потоков на улично-дорожной сети.

Плагин «ITSGIS. Дислокация дорожных знаков». Дает возможность нанесения дорожных знаков (как точечных объектов на карте) и хранения связанной информации (фото, ответственной организации, типа опоры и т.п.).

Плагин «ITSGIS. Дислокация светофоров». Позволяет наносить светофоры (как точечные объекты на карте) и хранения связанной информации (фото, параметров светофорного цикла, ответственной организации, даты монтажа, типа опоры и т.п.).

Плагин «ITSGIS. Дислокация ограждений». Обеспечивает автоматизированную дислокацию, редактирование и отображение на улично-дорожной сети пешеходных и транспортных ограждений.

Плагин «ITSGIS. Инфо». Предназначен для предоставления информационных услуг населению и представляет собой городской информационный справочник.

Плагин «ITSGIS. Дислокация остановок». Позволяет создавать и обрабатывать геообъекты транспортной инфраструктуры города – остановки общественного транспорта, для учета их на электронной карте города.

Плагин «ITSGIS. Маршруты». Предназначен для построения и поиска маршрутов с учетом различных критериев оптимальности, отображения их на электронной карте города.

Обзор систем-аналогов

QGIS (Quantum GIS) – свободная географическая информационная система с открытым кодом, которая предоставляет пользователям следующие возможности:

- просмотр данных – можно просматривать и накладывать друг на друга векторные и растровые данные в различных форматах и проекциях без преобразования во внутренний или общий формат;
- исследование данных и компоновка карт;
- управление данными: создание, редактирование и экспорт – в *QGIS* можно создавать и редактировать векторные данные, а также экспортировать их в разные форматы;

- анализ данных – *QGIS* предоставляет пользователям возможность анализировать векторные пространственные данные, использовать инструменты анализа, выборки, геопроессинга, управления геометрией и базами данных;
- публикация карт в сети Интернет – *QGIS* может использоваться для экспорта данных в *map*-файл и публикации его в сети Интернет, используя установленный веб-сервер *Mapserver*.

На рисунке 4 представлено главное окно системы *QGIS*.

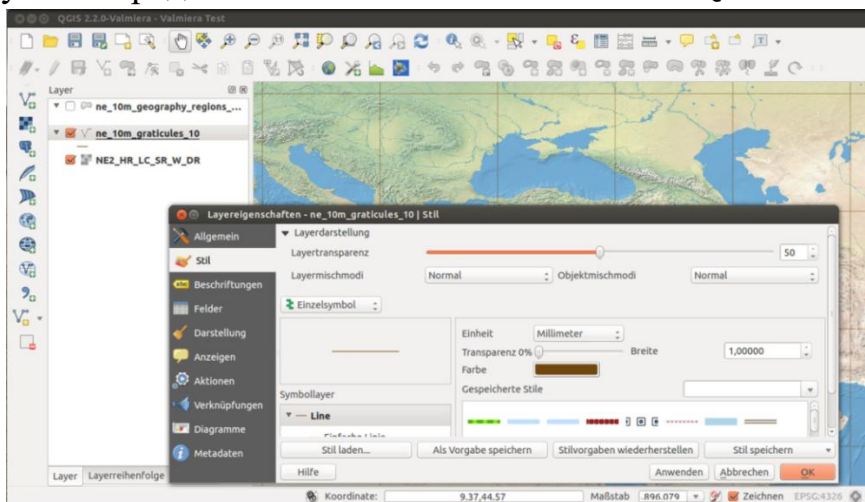


Рисунок 4. Главное окно системы *QGIS*

2ГИС – российская геоинформационная система со справочником карт. На рисунке 5 представлено главное окно системы 2ГИС.

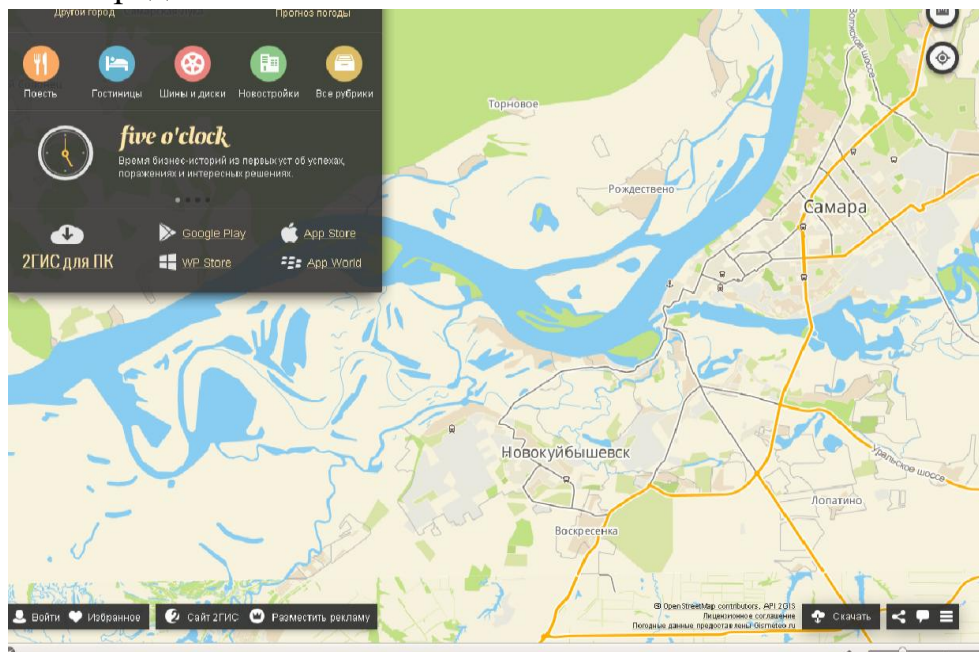


Рисунок 5. Главное окно системы 2ГИС

Основные преимущества данной системы:

- работа оффлайн, а также на мобильных платформах;

- детальное отображение карты;
- актуальная информация о фирмах города: контакты, время работы, отзывы, рейтинг, фотографии и входы;
- поиск проезда: оптимальные маршруты на авто и на общественном транспорте с подсказками по навигации;
- этажи: навигатор по крупным торговым центрам;
- справочник обслуживающих организаций.

Недостатками в данной системе являются большая зависимость от исходных географических данных, их точности и четкости их переноса в ИТСГИС, а также сложность анализа объектов.

Данные геоинформационные системы известны и широко используются, однако по сравнению с *ITSGIS* у них нет возможности отображать технические средства организации дорожного движения (ТСОДД) и характеристик транспортного потока:

- дорожные ограждения;
- дорожные знаки;
- дорожная разметка;
- столбы уличного освещения;
- остановки общественного транспорта;
- интенсивность дорожного движения (количество машин за единицу времени).

Постановка задачи

В ходе выполнения курсового проекта необходимо разработать адресный план города Владимира в геоинформационной системе *ITSGIS*.

Для решения поставленной задачи необходимо реализовать систему, выполняющую следующие функции:

- а) общесистемные:
 - авторизация пользователя в системе;
 - выбор слоя карты;
 - просмотр подробной информации об объектах на карте;
 - аутентификация пользователя в системе;
 - настройка интерфейса пользователя в зависимости от роли;
 - визуализация слоев;
 - визуализация карты;
 - перемещение по карте;
 - выбор/изменение масштаба карты;
 - выдача справочной информации о системе;
- б) функции редактора справочной информации:
 - ввод информации об объекте в базу данных;

- редактирование информации об объекте в базе данных;
- удаление информации об объекте из базы данных;
- в) функции редактора карты:
 - добавление объектов на слой карты;
 - редактирование объектов на слое карты;
 - удаление объектов на слое карты;
- г) функции редактора атрибутивной информации:
 - редактирование атрибутивной информации объектов на карте.

Описание логической модели базы данных

Логическая модель описывает понятия предметной области, их взаимосвязь, а также ограничения на данные, налагаемые предметной областью.

Логическая модель данных является начальным прототипом будущей базы данных. Она строится в терминах информационных единиц, но без привязки к конкретной СУБД. Более того, логическая модель данных обязательно должна быть выражена средствами именно реляционной модели данных.

Различают три уровня логической модели, отличающихся по глубине представления информации о данных:

- диаграмма сущность-связь (*Entity Relationship Diagram, ERD*);
- модель данных, основанная на ключах (*Key Based model, KB*);
- полная атрибутивная модель (*Fully Attributed model, FA*).

Диаграмма сущность-связь представляет собой модель данных верхнего уровня. Она включает сущности и взаимосвязи, отражающие основные бизнес-правила предметной области. Такая диаграмма не слишком детализирована, в нее включаются основные сущности и связи между ними, которые удовлетворяют основным требованиям, предъявляемым к системе. Диаграмма сущность-связь может включать связи многие-ко-многим и не включать описание ключей.

Модель данных, основанная на ключах, - более подробное представление данных. Она включает описание всех сущностей и первичных ключей и предназначена для представления структуры данных и ключей, которые соответствуют предметной области.

Полная атрибутивная модель - наиболее детальное представление структуры данных: представляет данные в третьей нормальной форме и включает все сущности, атрибуты и связи.

На рисунке 6 представлена логическая модель данных проектируемой системы, созданная в среде проектирования *Erwin Data Modeller*.

Основные компоненты диаграммы *Erwin* – это сущности, атрибуты и связи. Каждая сущность является множеством подобных индивидуальных объ-

ектов, называемых экземплярами. Каждый экземпляр индивидуален и должен отличаться от всех остальных экземпляров. Атрибут выражает определенное свойство объекта. Первичный ключ (*primary key*) – это атрибут или группа атрибутов, однозначно идентифицирующая экземпляр сущности. Ключи могут быть сложными, т. е. содержащими несколько атрибутов.

Для проектируемой системы была выбрана реляционная модель базы данных, как наиболее простая и подходящая кругу решаемых задач. Данные в базе представляются в табличной форме, на пересечении каждой строки и столбца таблицы находится только одно значение, все значения в каждом столбце имеют один тип, запросы к базе данных возвращают результат в табличной форме, что удобно для отображения и восприятия. Общепринятым стандартом языка работы с реляционными базами данных является язык *SQL*, который позволяет делать запрос на выдачу информации, объединенной по любому принципу.

Данные в базе данных представлены в нормализованной форме, т.е. устранена избыточность, которая, в свою очередь, потенциально приводит к различным аномалиям и нарушениям целостности данных.

В таблице 6 представлено описание фрагмента логической модели данных системы.

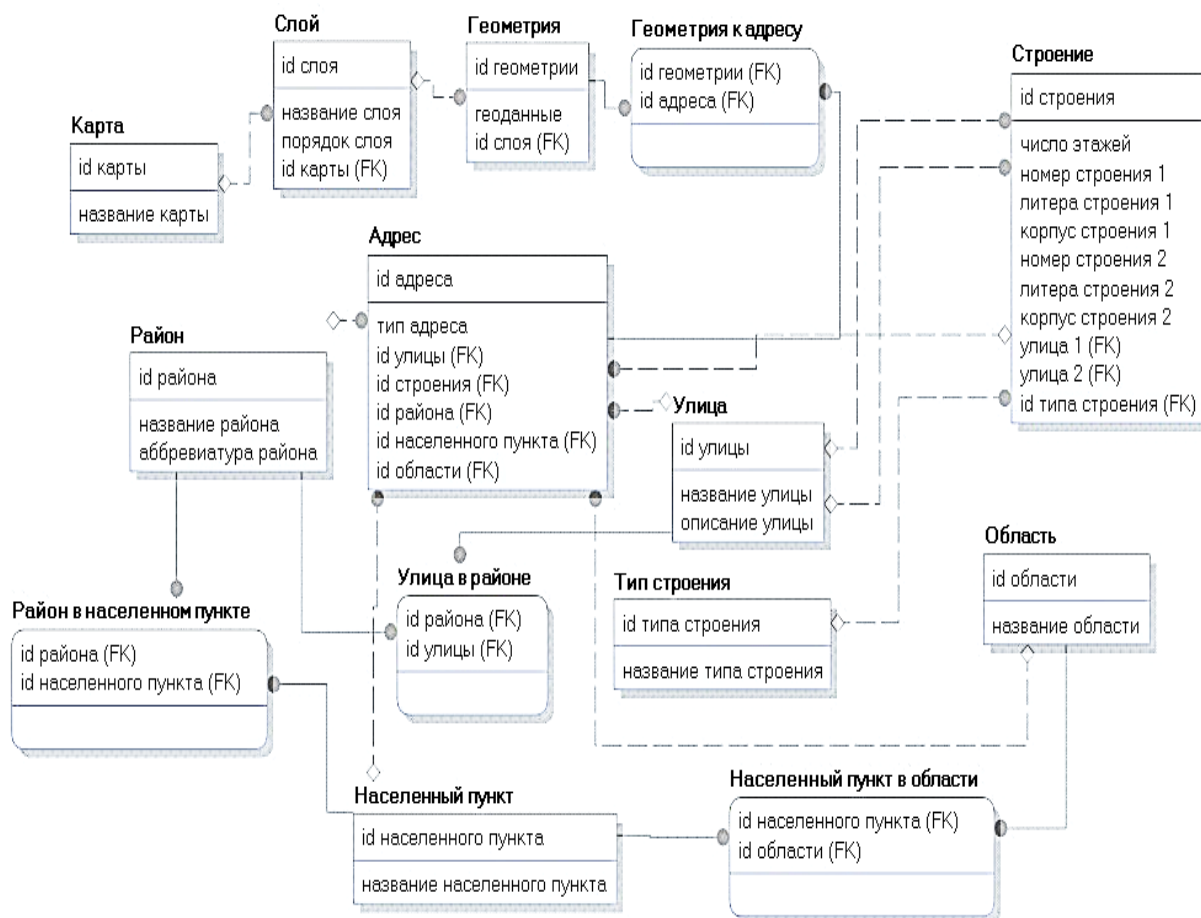


Рисунок 6. Фрагмент логической модели данных системы

Таблица 1. Описание фрагмента логической модели данных системы

Сущность	Описание
Строение	Содержит сведения о строении (число этажей, номер строения, литера, корпус, улица, тип строения)
Слой	Содержит сведения о слое карты (название, порядок, карта)
Геометрия	Содержит сведения о геометрии карты (геоданные, слой на карте)
Карта	Содержит сведения о карте (название)
Район	Содержит сведения о районе на карте (название, абривиатура)
Адрес	Содержит сведения об адресе на карте (тип, улица, строение, район, населенный пункт, область)
Улица	Содержит сведения об улице на карте (название, описание)
Область	Содержит сведения об области на карте (название)
Тип строения	Содержит сведения о типе строения на карте (название)
Населенный пункт	Содержит сведения о населенном пункте на карте (название)
Геометрия к адресу	Связывает сущности «Геометрия» и «Адрес», не содержит собственных атрибутов.
Район в населенном пункте	Связывает сущности «Район» и «Населенный пункт», не содержит собственных атрибутов.
Улица в районе	Связывает сущности «Улица» и «Район», не содержит собственных атрибутов.
Населенный пункт в области	Связывает сущности «Населенный пункт» и «Область», не содержит собственных атрибутов.

Алгоритм

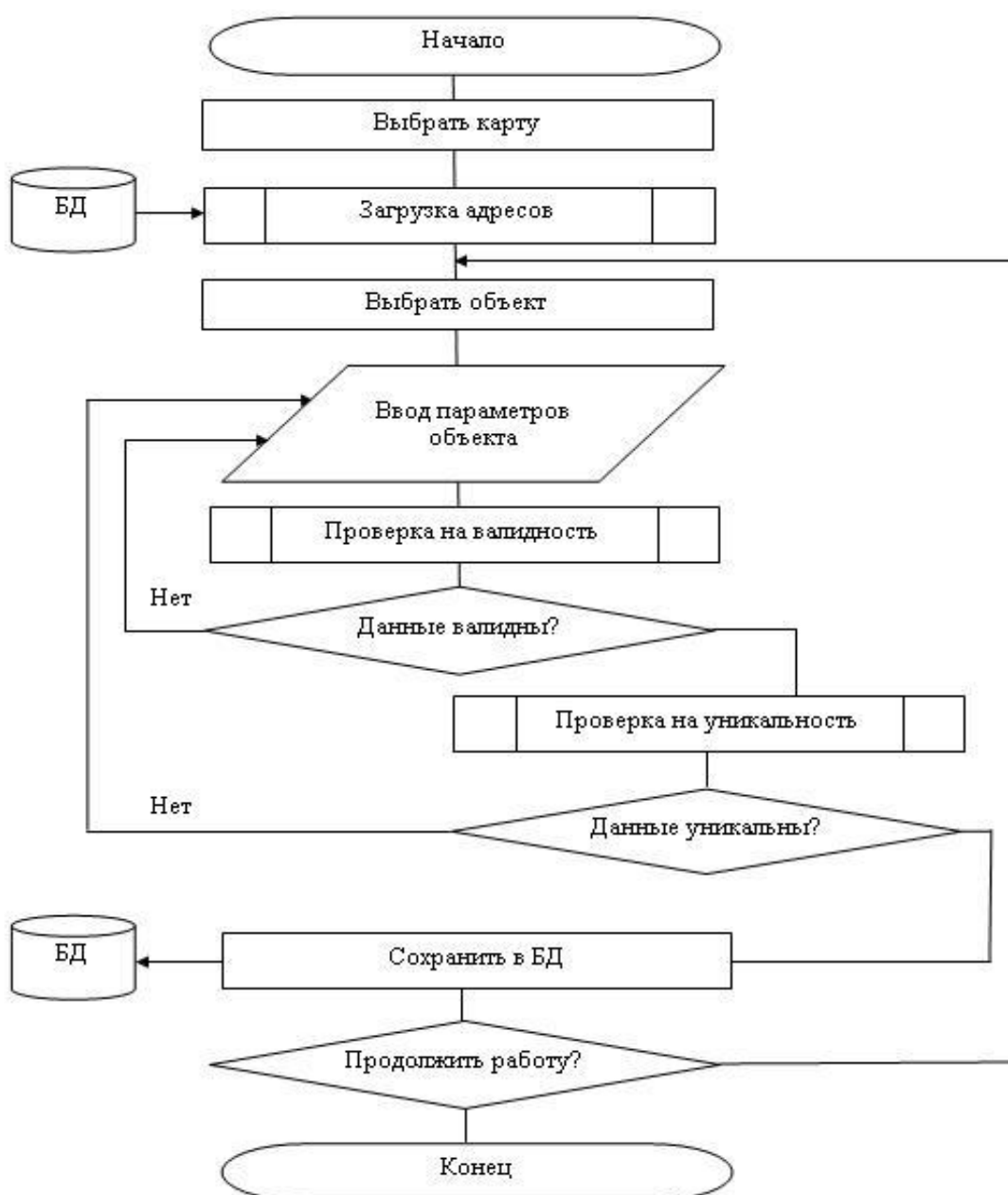


Рисунок 7. Алгоритм работы приложения

Выбор и обоснование комплекса программных средств

Программный комплекс – программа, состоящая из двух или более компонентов и (или) комплексов, выполняющих взаимосвязанные функции, и применяемая самостоятельно или в составе другого комплекса.

Выбор операционной системы

Операционная система играет роль посредника между пользователем, программами и оборудованием компьютера. Она обеспечивает возможность запуска программ, поддерживает работоспособность устройств, предоставляет средства проверки и настройки различных компонентов. Чем гибче опера-

ционная система, тем больше возможностей она предоставляет, тем удобнее работать с компьютером.

Операционная система *Windows* была выбрана, исходя из распространенности данной системы на персональных компьютерах. По состоянию на 2000 года под управлением операционных систем семейства *Windows* по данным ресурса *NetMarketShare (Net Applications)* работает около 90% персональных компьютеров. *Windows* привычна почти каждому обладателю персонального компьютера, имеет удобный и функциональный интерфейс. Программирование под эту систему пользуется популярностью, т.к. эта система распространена по всему миру.

Выбор языка программирования и среды разработки

В качестве языка программирования был выбран язык *C#*. Главной особенностью языка *C#* является его ориентированность на платформу *Microsoft.NET* – создатели *C#* ставили своей целью представление разработчикам естественных средств доступа ко всем возможностям платформы *.NET*. *.NET* платформа упрощает разработку приложений и повышает надежность кода. В частности, она обеспечивает автоматическое управление временем жизни объектов, нейтральность к языкам библиотеки классов и пересекающие границы языков, наследование, обработку исключений и отладку.

Таким образом, *C#* представляет собой новый язык программирования, ориентированный на разработку для платформы *.NET* и пригодный как для быстрого прототипирования приложений, так и для разработки крупномасштабных приложений.

Описание интерфейса

При запуске программы открывается форма авторизации пользователя, представленная на рисунке 8. Для входа в систему пользователь должен ввести в нее свой логин и пароль. После проверки данных происходит вход в систему или выдается сообщение об ошибке.

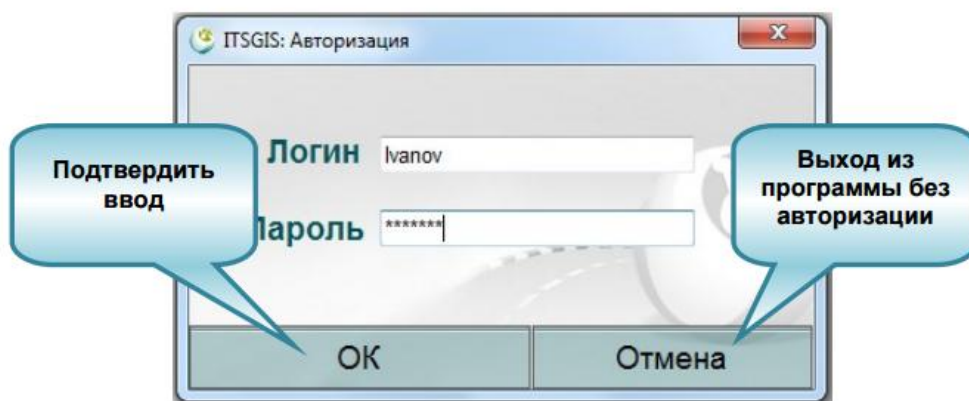


Рисунок 8. Форма авторизации пользователя

После того как авторизация будет пройдена, откроется окно для выбора открытых для пользователя городов, представленное на рисунке 9, с которым будет работать пользователь. Пользователю необходимо выбрать нужный ему город и подтвердить свой выбор.

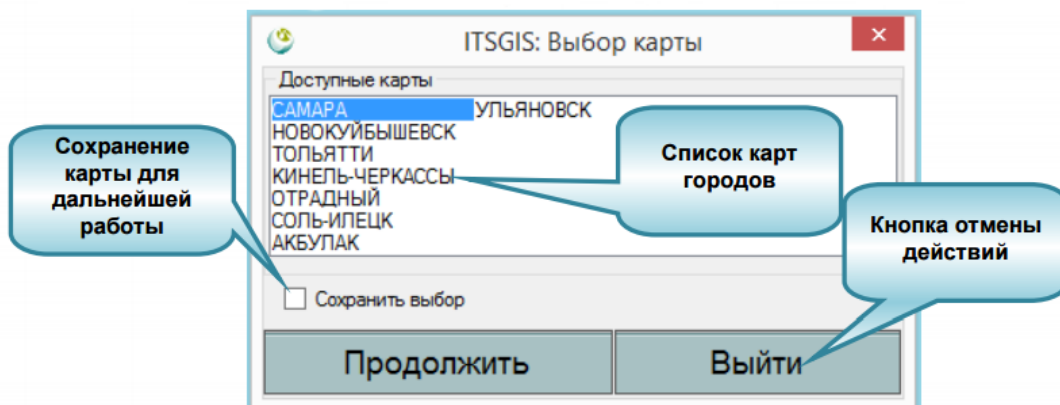


Рисунок 9. Окно для выбора города

После окончания загрузки, в окне главной формы появляется карта города и интерфейс программы, представленная на рисунке 10.

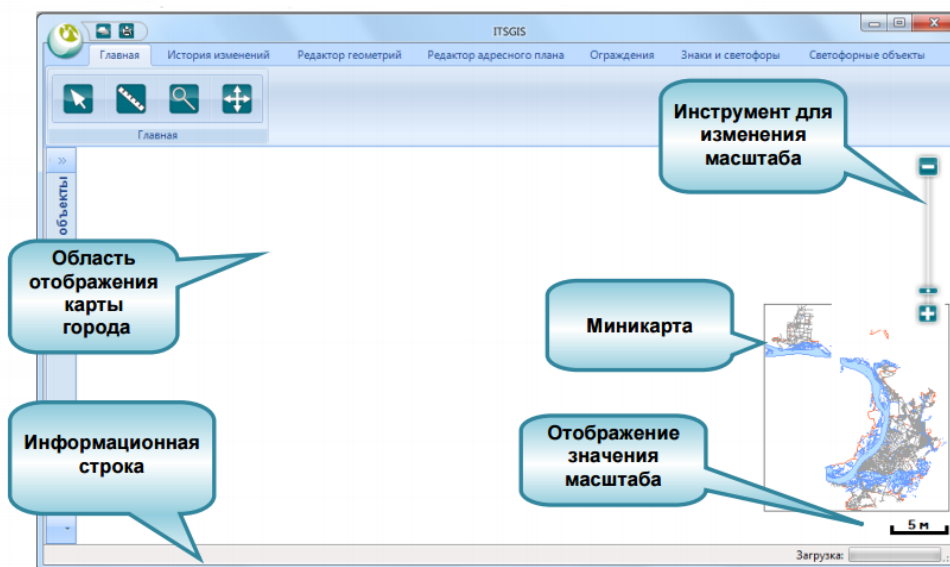


Рисунок 10. Главное окно программы

Пункт меню «Менеджер слоев» дает возможность перейти к списку слоев, в который включены все слои загруженной ранее карты (список слоев формируется программой автоматически). Состав слоев изменять нельзя, можно изменить только порядок их отображения на время текущего сеанса пользователя. Каждый слой имеет признак отображения слоя на карте, название и порядковый номер.

Работать с группой слоев не допускается, для работы с конкретным слоем необходимо выбрать его на панели, представленной на рисунке 11.

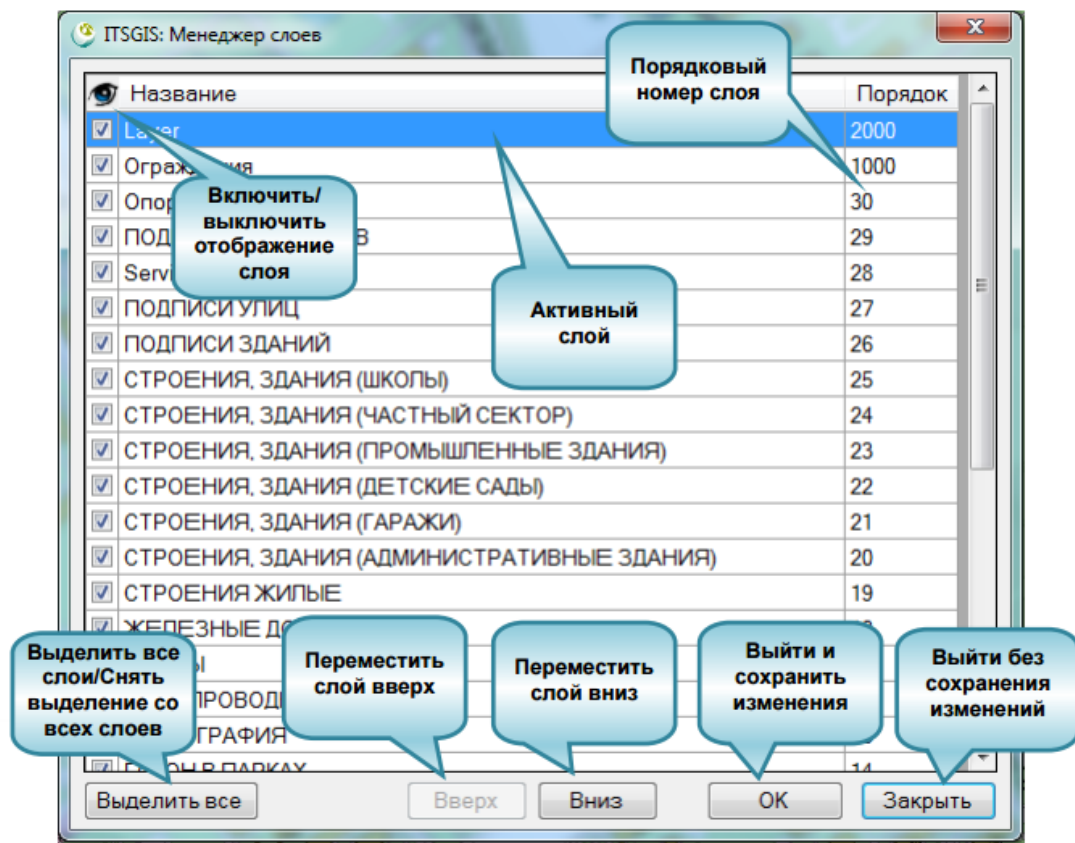


Рисунок 11. Панель работы со слоями

Для разработки адресного плана необходимо указать на слое «Подписи зданий» номера домов зданий, которым соответствуют геометрии на карте. Для этого пользователю надо воспользоваться инструментом «Добавить текст» на панели «Редактор геометрий», представленной на рисунке 12.

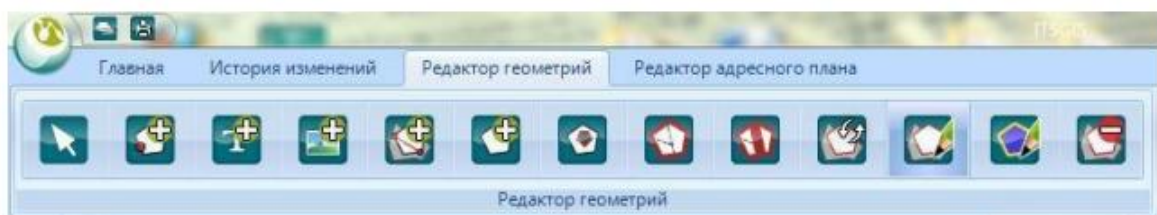


Рисунок 12. Панель "Редактор геометрии"

Далее, необходимо создать в базе данных соответствующие геометрии дома. Сделать это можно на панели «Редактор адресного плана», представленном на рисунке 13.

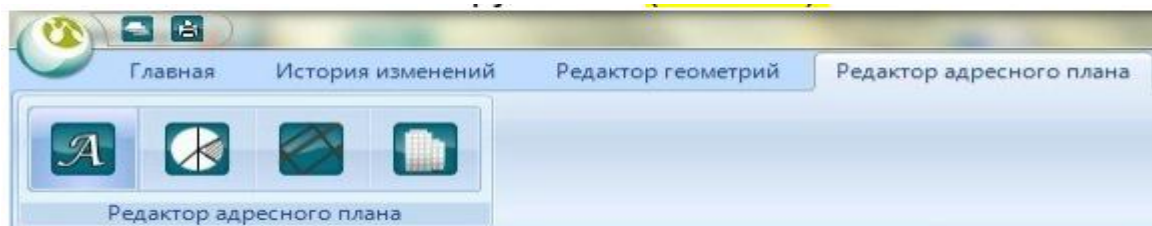


Рисунок 13. Панель «Редактор адресного плана»

Выбрав на этой панели инструмент «Редактировать справочники адресного плана», пользователь откроет окно редактирования справочников адресного плана города, представленное на рисунке 14.

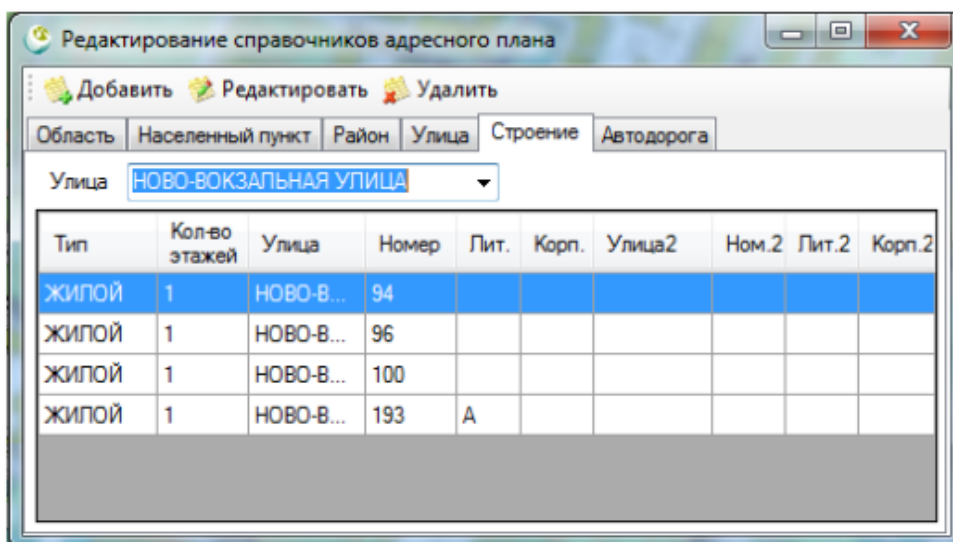


Рисунок 14. Окно «Редактирование справочников адресного плана»

В этом окне пользователь может создавать, удалять и редактировать объекты справочника адресного плана.

После того, как пользователь заполнит справочник адресного плана, ему необходимо связать объект в базе данных с геометрией на карте. Для этого на панели «Редактор адресного плана» выбираем инструмент «Связывание строения и геометрии», представленный на рисунке 15, после чего в открывшемся окне пользователь выбирает геометрию и соответствующий ей адрес из справочника адресов и добавляем ее к адресу.

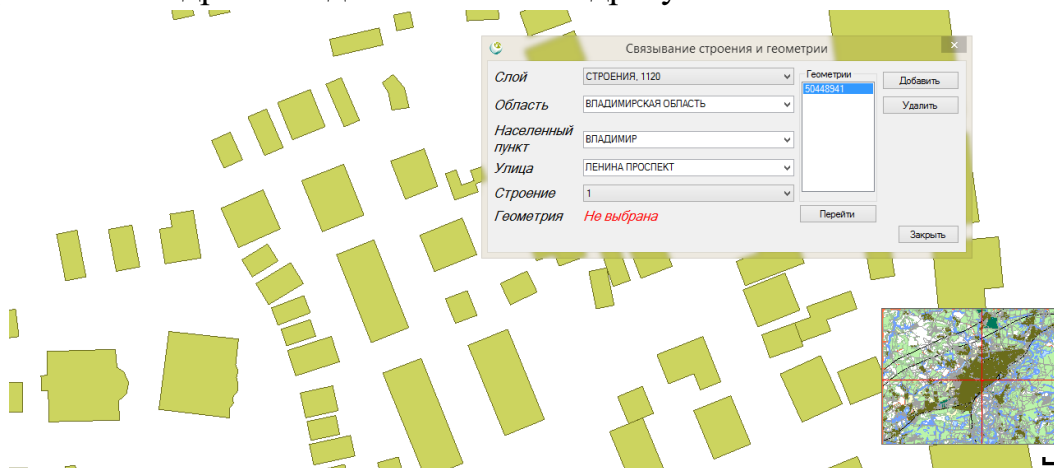


Рисунок 15. Окно «Связывание строения и геометрии»

Описание физической модели базы данных

Физическая модель базы данных описывает основные понятия предметной области, а также их взаимосвязь с учетом реализации конкретной системы управления базами данных. Специфика конкретной СУБД может вклю-

чать в себя ограничения на именование объектов базы данных, на поддерживаемые типы данных, на реализацию ограничений, имеющихся в логической модели данных и другие.

В таблице 2 представлено описание фрагмента физической модели данных системы.

Таблица 2. Описание фрагмента физической модели данных системы

Map			
id_map	integer	primary key	not null
name_map	varchar(30)		not null
Layer			
id_layer	integer	primary key	not null
name_layer	varchar(30)		not null
order_layer	integer		not null
id_map	integer	foreign key	not null
Geometry			
id_geometry	integer	primary key	not null
geodata	varchar(30)		not null
id_layer	integer	foreign key	not null
Geometry_to_address			
id_geometry	integer	foreign key	not null
id_address	integer	foreign key	not null
Building			
id_building	integer	primary key	not null
number_floors	integer		null
number_building1	integer		not null
lit_building1	char		null
korp_building1	integer		null
number_building2	integer		null
lit_building2	char		null
korp_building 2	integer		null
id_street1	varchar(30)	foreign key	not null
id_street2	varchar(30)	foreign key	null
id_type	integer	foreign key	not null
Address			
id_address	integer	primary key	not null

address_type	varchar(30)		not null
id_street	integer	foreign key	not null
id_building	integer	foreign key	not null
id_locality	integer	foreign key	not null
id_region	integer	foreign key	not null
id_area	integer	foreign key	not null
Area			
id_area	integer	primary key	not null
name_area	varchar(30)		not null
abb_area	varchar(30)		not null
Street			
id_street	integer	primary key	not null
name_street	varchar(30)		not null
about_street	varchar(30)		null
Region			
id_region	integer	primary key	not null
name_region	varchar(30)		not null
Street_to_area			
id_street	integer	foreign key	not null
id_area	integer	foreign key	not null
Area_to_locality			
id_area	integer	foreign key	not null
id_locality	integer	foreign key	not null
Locality			
id_locality	integer	primary key	not null
name_locality	varchar(30)		not null
Locality_to_region			
id_locality	integer	foreign key	not null
id_region	integer	foreign key	not null
building_type			
id_type	integer	primary key	not null
name_type	varchar(30)		not null

На рисунке 16 представлена физическая модель базы данных.



В соответствии с техническим заданием был разработан адресный план городов России в геоинформационной системе *ITSGIS*.

Во втором разделе описаны этапы проектирования системы: была разработана структурная схема системы, описана спецификация требований, проведено прототипирование интерфейса пользователя системы. Также произведен выбор и обоснование комплекса программных средств.

В третьем разделе описана реализация системы: интерфейс системы и физическая модель базы данных системы.

1. Географическая информационная система (ГИС) . – URL: <http://moslesproekt.roslesinfor.org.ru/activity/023gil-inform> (дата обращения 10.10.2024);
2. Википедия. Геоинформационная система. – URL: https://ru.wikipedia.org/wiki/геоинформационная_система (дата обращения 10.10.2024).

3. Геоинформационная система ITSGIS. – URL: <http://www.itsgis.ru> (дата обращения 10.10.2024 г.).
4. Геоинформационная система 2ГИС. – URL: <https://ru.wikipedia.org/wiki/2ГИС> (дата обращения 10.10.2024 г.).
5. Михеев, С.В. Интеллектуальная транспортная геоинформационная система ITSGIS / Т.И. Михеева, С.В. Михеев, О.К. Головнин // Современные проблемы безопасности жизнедеятельности: интеллектуальные транспортные системы : материалы IV Международной научно-практической конференции (Казань, 25–26 февраля 2016 г.). – Казань : ГБУ «Научный центр безопасности жизнедеятельности», 2016. – С. 362–368. – ISBN 978-5-85247-837-5.
6. Головнин, О.К. Универсальное программное средство для геоинформационных систем сенсорных инфокиосков / Т.И. Михеева, С.В. Михеев, О.К. Головнин // Региональная научно-практическая конференция, посвященная 50-летию первого полета человека в космос : тезисы докладов. – Самара : Издательство Самарского государственного аэрокосмического университета, 2011. – С. 235–236. – ISBN 978-5-7883-0839-5.
7. Golovnin, O.K. Interactive content management in the system “Infomatriks” / T.I. Mikheeva, S.V. Mikheev, O.K. Golovnin, I.G. Bogdanova // Proceedings of the 14th international workshop on computer science and information technologies CSIT’2012, Ufa – Hamburg – Norwegian Fjords, September 20–26, 2012. – P. 285–289.
8. Смолев, А.М. Методы моделирования адресного плана в рамках геоинформационной системы / А.М. Смолев, Т.И. Михеева, А.В. Золотовицкий // IT & Transport / ИТ & Транспорт : Сборник научных статей. – Самара : ИнтелТранС, 2023. – Т. 24. – С. 4-13.
9. Михеева, Т.И. Информационная технология автоматической дислокации геообъектов транспортной инфраструктуры на улично-дорожной сети / Т.И. Михеева, А.В. Сидоров, О.К. Головнин // Перспективные информационные технологии : труды Международной научно-технической конференции, Самара, 04–06 декабря 2013 года / Самарский государственный аэрокосмический университет им. академика С.П. Королева. – Самара : Самарский научный центр РАН, 2013. – С. 236-241.
10. Михеева, Т.И. Инструментальная среда для проектирования объектов интеллектуальной транспортной системы / Т.И. Михеева // Вестник Самарского гос. техн. ун-та. Серия «Технические науки» №40. Самара: СамГТУ, – 2006. – С.96-103.
11. Интеллектуальная транспортная геоинформационная система ITSGIS. Ядро / Т.И. Михеева, С.В. Михеев, О.К. Головнин, А.В. Си-

- доров, Е.А. Савинов. – Самара : Интелтранс, 2016. – Т.1. – 171 с. – ISBN 978-5-9906857-4-1.
- 12.Малыгин, И.Г. Сети, информация и знания – основные драйверы Четвертой индустриальной революции (INDUSTRIE 4.0) / И.Г. Малыгин, В.И. Комашинский, М.Ю. Аванесов [и др.] // Информация и космос. – 2016. – № 1. – С. 14-25.
- 13.Гончаров, Е.Ю. Микросервисная архитектура в контексте разработки интеллектуальных транспортных систем / Е.Ю. Гончаров // Сборник лучших докладов студенческой научно-технической конференции, посвященной 100-летию Отечественной гражданской авиации : Сборник материалов конференции, Московский государственный технический университет гражданской авиации, 13 апреля 2023 года. – Москва: ИД Академии Н.Е. Жуковского, 2023. – С. 77-81.

Mikheeva T.I., Smolev A.M.

**DEVELOPMENT OF ADDRESS PLANS FOR RUSSIAN
CITIES IN INTELLIGENT TRANSPORT GEOGRAPHIC
INFORMATION SYSTEM «ITSGIS»**

*Samara, Samara University named after academician S.P. Korolev
IntelTrans*

This article presents the transport problems of modern Russian cities during the intelligent transport geoinformation system "ITSGIS". A system has been developed that allows them to be extremely widely used and have a serious scientific basis.

Keywords: design, street and road network, cities, intensity, intelligent transport geoinformation system.

СПИСОК АВТОРОВ СБОРНИКА

Михеева Татьяна Ивановна, доктор технических наук, профессор кафедры информационных систем и технологий Самарского университета (национального исследовательского университета), генеральный директор группы компаний «ИнтелТранС» (ITSGIS).

Михеев Сергей Владиславович, кандидат технических наук, доцент кафедры организации и управления перевозками на транспорте Самарского университета (национального исследовательского университета).

Смолев Александр Михайлович, диссертация Самарского университета им. академика С.П. Королева (национального исследовательского университета), программист отдела программирования научно-производственного центра «Интеллектуальные транспортные системы».

Клепиков Никита Михайлович, диссертация Самарского университета им. академика С.П. Королева (национального исследовательского университета), кафедра технической кибернетики института информатики, математики и электроники.

СОДЕРЖАНИЕ

<i>Михеева Т.И., Клепиков Н.М. КЛАССИФИКАЦИЯ БЕСПИЛОТНЫХ ЛЕТАТЕЛЬНЫХ АППАРАТОВ.....</i>	<i>4</i>
<i>Михеева Т.И. СОВРЕМЕННЫЕ МЕТОДЫ МИКРОМОДЕЛИРОВАНИЕ ДВИЖЕНИЯ ТРАНСПОРТНЫХ СРЕДСТВ В «ITSGIS»: РАЗРАБОТКИ И ПРОЕКТИРОВАНИЯ СЛОЖНЫХ ПРОГРАММНЫХ КОМПЛЕКСОВ....</i>	<i>13</i>
<i>Михеева Т.И., Смолев А.М. РАЗРАБОТКА АДРЕСНЫХ ПЛАНОВ ГОРОДОВ РОССИИ В ИНТЕЛЛЕКТУАЛЬНОЙ ТРАНСПОРТНОЙ ГЕОИНФОРМАЦИОННОЙ СИСТЕМЫ «ITSGIS»</i>	<i>47</i>

Сетевое издание

IT & TRANSPORT / ИТ & ТРАНСПОРТ

Редакционная коллегия

Главный редактор:

Михеева Татьяна Ивановна – доктор технических наук, профессор, заслуженный деятель науки и образования РФ», (Самара)

Заместители главного редактора:

Прохоров Сергей Антонович – д.т.н., заместитель генерального директора ООО «НПЦ «ИТС», заслуженный работник высшей школы РФ, академик Академии навигации и управления движением, доктор технических наук, профессор, (Самара),

Филиппова Анна Сергеевна – д.т.н., «Башкирский государственный педагогический университет им. М. Акмуллы», (Уфа)

Секретарь редакционной коллегии:

Чекина Елена Владимировна – ООО «НПЦ «ИТС», (Самара)

Редакционный совет:

Бурдин Антон Владимирович – д.т.н., «Санкт-Петербургский Государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича», (Санкт-Петербург),

Гераськин Михаил Иванович – д.э.н., Самарский университет, (Самара),

Жанказиев Султан Владимирович – д.т.н., интеллектуальные транспортные системы МАДИ, (Москва),

Заболотнов Юрий Михайлович – д.т.н., Самарский университет, (Самара),

Зырянов Владимир Васильевич – д.т.н., РГСУ, (Ростов-на-Дону),

Иващенко Антон Владимирович – д.т.н., Передовая медицинская инженерная школа Самарского государственного медицинского университета Министерства здравоохранения Российской Федерации, (Самара),

Минниханов Рифкат Нургалиевич – д.т.н., ГБУ «Безопасность дорожного движения», (Казань),

Хайтбаев Валерий Абдурахманович – д.э.н., Самарский университет, (Самара), Самарский университет, (Самара),

Зеленко Лариса Сергеевна – к.т.н., Самарский университет, (Самара),

Золотовицкий Аркадий Владимирович – к.т.н., Самарский университет, (Самара),

Михеев Сергей Владиславович – к.т.н., ООО «Научно-производственный центр «Интеллектуальные транспортные системы», (Самара),

Сапрыкин Олег Николаевич – к.т.н., Самарский университет, (Самара),

Сапрыкина Ольга Валерьевна – к.т.н., Самарский университет, (Самара),

Федосеев Александр Андреевич – к.т.н., ОАО «РКУ «Прогресс», (Самара),

Шопин Андрей Геннадьевич – к.т.н., ООО «СМС-информационные технологии», (Самара).

Издательство ООО «НПЦ «ИТС»
443099, Самарская обл., г.о. Самара, г. Самара,
вн. р-н Самарский, ул. Куйбышева, д. 71, оф. 2
Подписано к выпуску 17.12.2024.
Распространяется бесплатно

Учредитель
ООО «НПЦ «ИТС»
Адрес редакции журнала «ИТ & Транспорт»
443125, г. Самара, пр. Кирова, 328, 67