

Министерство образования и науки Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Школа: ИШИТР

Направление подготовки (специальность): 09.03.04 «Программная инженерия»

Отделение школы (НОЦ): ИТ

БАКАЛАВРСКАЯ РАБОТА

Тема работы
Разработка подсистем видеонаблюдения и мониторинга дорожной ситуации в автоматизированной системе управления дорожного движения

УДК: 004.78:681.772.7:656

Студент

Группа	ФИО	Подпись	Дата
8К4Б	Нечухин А.В.		

Руководитель

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель отделения ИТ	Мокина Е.Е.			

КОНСУЛЬТАНТЫ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент	Авдеева И.И.			

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент отделения соц. гуманитарных наук	Петухов О.Н.	канд. экон. наук		

ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент отделения информационных технологий	Чердынцев Е.С.	канд. техн. наук		

Томск – 2018 г.

Школа: ИШИТР

Направление подготовки (специальность): 09.03.04 «Программная инженерия»

Отделение школы (НОЦ): ИТ

УТВЕРЖДАЮ:

Руководитель ООП

_____ Чердынцев Е.С.
(Подпись) (Дата) (Ф.И.О.)

ЗАДАНИЕ

на выполнение выпускной квалификационной работы

В форме:

Бакалаврской работы

(бакалаврской работы, дипломного проекта/работы, магистерской диссертации)

Студенту:

Группа	ФИО
8К4Б	Нечухину А.В.

Тема работы:

Разработка подсистем видеонаблюдения и мониторинга дорожной ситуации в автоматизированной системе управления дорожного движения

Утверждена приказом директора (дата, номер)

Срок сдачи студентом выполненной работы:

18 июня 2018 г.

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

Исходные данные к работе

(наименование объекта исследования или проектирования; производительность или нагрузка; режим работы (непрерывный, периодический, циклический и т. д.); вид сырья или материал изделия; требования к продукту, изделию или процессу; особые требования к особенностям функционирования (эксплуатации) объекта или изделия в плане безопасности эксплуатации, влияния на окружающую среду, энергозатратам; экономический анализ и т. д.).

1. Спецификация, определяющая передачу данных в промышленных сетях и взаимодействие устройств в них (OPC Unified Architecture)
2. Техническое задание на разработку подсистемы интеллектуального видеонаблюдения AM.Eye
3. Программное обеспечение для разработки, визуализации и управления объектами технологического процесса на мнемосхемах в реальном масштабе времени (InfinityHMI, входящее в состав SCADA Infinity, разработанное АО ЭлеСи)
4. Техническое задание на разработку подсистемы мониторинга транспортных потоков AM.Traffic
5. Описание детекторов, разработанных компанией SmartMicro (UMRR-0C Type 39, Type 42)
6. Программный интерфейс приложения, предоставленный компанией СОРБ Групп для получения информации с датчиков SmartMicro

	7. Платформа, реализующая систему обмена сообщениями между компонентами программной системы (RabbitMQ)
<p>Перечень подлежащих исследованию, проектированию и разработке вопросов</p> <p><i>(аналитический обзор по литературным источникам с целью выяснения достижений мировой науки техники в рассматриваемой области; постановка задачи исследования, проектирования, конструирования; содержание процедуры исследования, проектирования, конструирования; обсуждение результатов выполненной работы; наименование дополнительных разделов, подлежащих разработке; заключение по работе).</i></p>	<p>1. Общее описание всех подсистем автоматизированной системы управления дорожного движения</p> <p>2. Обзор производителей систем видеонаблюдения</p> <p>3. Разработка подсистемы видеонаблюдения</p> <p>4. Разработка подсистемы мониторинга транспортных потоков</p> <p>3. Изучение и совместная доработка API, предоставленного компанией СОРБ Групп</p> <p>4. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</p> <p>5. Социальная ответственность</p>
<p>Перечень графического материала</p> <p><i>(с точным указанием обязательных чертежей)</i></p>	
<p>Консультанты по разделам выпускной квалификационной работы</p> <p><i>(с указанием разделов)</i></p>	
Раздел	Консультант
1. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Авдеева И.И.
2. Социальная ответственность	Петухов О.Н.

Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	
---	--

Задание выдал руководитель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель отделения ИТ	Мокина Е.Е.			

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8К4Б	Нечеухин А.В.		

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»**

Студенту:

Группа	ФИО
8К4Б	Нечеухину А.В.

Школа	ИШИТР	Отделение школы	ИТ
Уровень образования	Бакалавриат	Направление подготовки	09.03.04 Программная инженерия

Исходные данные к разделу «Социальная ответственность»

1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	Разработка подсистем видеонаблюдения и мониторинга дорожной ситуации в автоматизированной системе управления дорожного движения с использованием технических средств (персональный компьютер, монитор, мышь компьютерная, клавиатура). Основная область применения разрабатываемой системы – управление магистральным движением.
--	---

Перечень вопросов, подлежащих исследованию, проектированию и разработке

<p>1. Производственная безопасность:</p> <p>1.1. Анализ вредных факторов проектируемой производственной среды в следующей последовательности:</p> <ul style="list-style-type: none"> - физико-химическая природа вредности, ее связь с разрабатываемой темой; - действие фактора на организм человека; - приведение допустимых норм с необходимой размерностью (со ссылкой на соответствующий нормативно-технический документ); - предлагаемые средства защиты (сначала коллективной защиты, затем – индивидуальные защитные средства). <p>1.2. Анализ опасных факторов проектируемой производственной среды в следующей последовательности:</p> <ul style="list-style-type: none"> - термические опасности (источники, средства защиты); 	<p>К вредным факторам можно отнести следующие факторы:</p> <ul style="list-style-type: none"> - микроклимат, который регламентируется ГОСТ 30494-2011 (Параметры микроклимата в помещениях), а также СанПиН 2.2.4.548-96 (Гигиенические требования к микроклимату производственных помещений); - освещенность, которая регламентируется ГОСТ Р 55710-2013 (Освещение рабочих мест внутри зданий), а также СанПиН 2.2.1/2.1.1.1278-03 (Гигиенические требования к естественному, искусственному и совмещенному освещению жилых и общественных зданий); - шум, который регламентируется ГОСТ 12.1.003-2014 (Шум. Общие требования безопасности), а также СН 2.2.4/2.1.8.562-96 (Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории жилой застройки). - умственное перенапряжение, которое регламентируется ТОИ Р-45-084-01 <p>Основным опасным фактором при работе на ПЭВМ является опасность поражения электрическим током, которая регламентируется ГОСТ Р 12.1.019-2009</p>
---	--

<ul style="list-style-type: none"> - электробезопасность (в т.ч. статическое электричество, молниезащита – источники, средства защиты); - пожаровзрывобезопасность (причины, профилактические мероприятия, первичные средства пожаротушения). 	(Электробезопасность. Общие требования и номенклатура видов защиты)
<p>2. Экологическая безопасность:</p> <ul style="list-style-type: none"> - анализ воздействия объекта на атмосферу (выбросы); - анализ воздействия объекта на гидросферу (сбросы); - анализ воздействия объекта на литосферу (отходы). 	- безопасный сбор, хранение, транспортирование и разработка отработавшего электротехнического и электронного оборудования за исключением ртутьсодержащих устройств и приборов по ГОСТ Р 55102-2012.
<p>3. Безопасность в чрезвычайных ситуациях:</p> <ul style="list-style-type: none"> - перечень возможных ЧС при разработке и эксплуатации проектируемого решения; - выбор наиболее типичной ЧС; - разработка превентивных мер по предупреждению ЧС; - разработка действий в результате возникшей ЧС и мер по ликвидации ее последствий. 	Наиболее типичная чрезвычайная ситуация при работе в офисе – пожар на рабочем месте.
<p>4. Правовые и организационные вопросы обеспечения безопасности:</p> <ul style="list-style-type: none"> - специальные правовые нормы трудового законодательства; - организационные мероприятия при компоновке рабочей зоны. 	<ul style="list-style-type: none"> - гигиенические требования к персональным электронно-вычислительным машинам и организации работы регламентируется СанПиН 2.2.2/2.4.1340-03; - трудовой кодекс Российской Федерации № 197-ФЗ

Дата выдачи задания для раздела по линейному графику	
---	--

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент	Авдеева И.И.			

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8К4Б	Нечеухин А.В.		

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И
РЕСУРСОСБЕРЕЖЕНИЕ»**

Студенту:

Группа	ФИО
8К4Б	Нечеухину А.В.

Школа	ИШИТР	Отделение школы	ИТ
Уровень образования	Бакалавриат	Направление подготовки	09.03.04 Программная инженерия

Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

1. Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих.	Работа направлена на разработку подсистем видеонаблюдения и мониторинга дорожной ситуации в автоматизированной системе управления дорожного движения.
2. Нормы и нормативы расходования ресурсов	
3. Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования	

Перечень вопросов, подлежащих исследованию, проектированию и разработке

1. Оценка коммерческого потенциала, перспективности проведения НИ с позиции ресурсоэффективности и ресурсосбережения	Анализ конкурентных решений, SWOT-анализ
2. Планирование и формирование бюджета научных исследований	Планирование этапов проведения работ, бюджет научно-технического исследования.

Перечень графического материала

1. Временные показатели проведения различных этапов работ
2. Линейный график работ

Дата выдачи задания для раздела по линейному графику	
---	--

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент отделения соц. гуманитарных наук	Петухов О.Н.	канд. экон. наук		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8К4Б	Нечеухин А.В.		

Министерство образования и науки Российской Федерации
 федеральное государственное автономное образовательное учреждение
 высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
 ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Школа: ИШИТР

Направление подготовки (специальность): 09.03.04 «Программная инженерия»

Уровень образования: бакалавриат

Отделение школы (НОЦ): ИТ

Период выполнения: осенний / весенний семестр 2017/2018 учебного года

Форма представления работы:

Бакалаврская работа

(бакалаврская работа, дипломный проект/работа, магистерская диссертация)

**КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН
 выполнения выпускной квалификационной работы**

Срок сдачи студентом выполненной работы:	18 июня 2018 г.
--	-----------------

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
20.04.2018	<i>Раздел 1. Обзор предметной области</i>	
30.04.2018	<i>Раздел 2. Разработка подсистемы видеонаблюдения</i>	
10.05.2018	<i>Раздел 3. Разработка подсистемы мониторинга транспортных потоков</i>	
20.05.2018	<i>Раздел 4. Социальная ответственность</i>	
31.05.2018	<i>Раздел 5. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</i>	

Составил преподаватель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель отделения ИТ	Мокина Е.Е.			

СОГЛАСОВАНО:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент отделения информационных технологий	Чердынцев Е.С.	канд. техн. наук		

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ООП

Код результата	Результат обучения	Требования ФГОС, критерии АИОР
Р1	Применять базовые и специальные естественнонаучные и математические знания в области информатики и вычислительной техники, достаточные для комплексной инженерной деятельности.	Требования ФГОС (ОК-1, 10, ПК-4, 5, 6), критерий 5 АИОР (п. 1.1)
Р2	Применять базовые и специальные знания в области современных информационных технологий для решения инженерных задач.	Требования ФГОС (ОК-11, 12, 13, ПК-1, 2, 11), критерий 5 АИОР (п.1.1, 1.2)
Р3	Ставить и решать задачи комплексного анализа, связанные с созданием аппаратно-программных средств информационных и автоматизированных систем, с использованием базовых и специальных знаний, современных аналитических методов и моделей.	Требования ФГОС (ОК-1, 8, ПК-2, 4, 6), критерий 5 АИОР (п. 1.2)
Р4	Разрабатывать программные и аппаратные средства (системы, устройства, блоки, программы, базы данных и т. п.) в соответствии с техническим заданием и с использованием средств автоматизации проектирования.	Требования ФГОС (ОК-2, 3, ПК-3, 4, 5), критерий 5 АИОР (п. 1.3)
Р5	Проводить теоретические и экспериментальные исследования, включающие поиск и изучение необходимой научно-технической информации, математическое моделирование, проведение эксперимента, анализ и интерпретация полученных данных, в области создания аппаратных и программных средств информационных и автоматизированных систем.	Требования ФГОС (ОК-6, ПК-6, 7), критерий 5 АИОР (п.1.4)
Р6	Внедрять, эксплуатировать и обслуживать современные программно-аппаратные комплексы, обеспечивать их высокую	Требования ФГОС (ОК-4,

	эффективность, соблюдать правила охраны здоровья, безопасность труда, выполнять требования по защите окружающей среды.	15, 16, ПК-9, 10, 11), критерий 5 АИОР (п. 1.5)
P7	Использовать базовые и специальные знания в области проектного менеджмента для ведения комплексной инженерной деятельности.	Требования ФГОС (ОК-1, 4, ПК-1, 6, 7), критерий 5 АИОР (п. 2.1)
P8	Владеть иностранным языком на уровне, позволяющем работать в иноязычной среде, разрабатывать документацию, презентовать и защищать результаты комплексной инженерной деятельности.	Требования ФГОС (ОК-14, ПК-7), критерий 5 АИОР (п. 2.2)
P9	Эффективно работать индивидуально и в качестве члена группы, состоящей из специалистов различных направлений и квалификаций, демонстрировать ответственность за результаты работы и готовность следовать корпоративной культуре организации.	Требования ФГОС (ОК-2, 3, 4), критерий 5 АИОР (п. 2.3, 2.4)
P10	Демонстрировать знания правовых, социальных, экономических и культурных аспектов комплексной инженерной деятельности.	Требования ФГОС (ОК-1, 5, 9), критерий 5 АИОР (п. 2.5)
P11	Демонстрировать способность к самостоятельной к самостоятельному обучению в течение всей жизни и непрерывному самосовершенствованию в инженерной профессии.	Требования ФГОС (ОК-6, 7), критерий 5 АИОР (п. 2.6)

Реферат

Выпускная квалификационная работа содержит: 102 страницы, 3 рисунка, 17 таблиц, 32 источника литературы, 4 приложения.

Ключевые слова: COM, SCADA Infinity, HMI, Milestone, подсистема видеонаблюдения, подсистема мониторинга, транспортные потоки, АСУДД.

Объектом исследования являются автоматизированные системы управления дорожного движения.

Целью работы является разработка подсистем, позволяющих производить мониторинг дорожной ситуации и различных параметров транспортного потока, а также оперативное оповещение участников дорожного движения.

В результате исследования были разработаны подсистемы видеонаблюдения и мониторинга транспортных потоков.

Степень внедрения: частичная договоренность о внедрении с ГК «Российские автомобильные дороги».

Область применения разработки: система может применяться в будущем на любом предприятии, где необходимо осуществление видеонаблюдения, а также основным заинтересованным лицом - ГК «Автодор» по мере реализации инновационных проектов.

Перечень условных обозначений, единиц и терминов

АДМС – автоматическая дорожная метеостанция.

АМ – программный комплекс «Автомагистраль».

АСУДД (Автоматизированная Система Управления Дорожного Движения - комплекс программных, организационных и технических мер, которые обеспечивают сбор и дальнейшую обработку информации о всех параметрах потоков транспорта.

ЗПИ – знак переменной информации.

ИТС (Интеллектуальная Транспортная Система) – это интеллектуальная система, использующая инновационные разработки в моделировании транспортных систем и регулировании транспортных потоков, предоставляющая конечным потребителям большую информативность и безопасность.

ПО – программное обеспечение.

ТПИ – табло переменной информации.

СОМ (Component Object Model) –технологический стандарт от компании Microsoft, предназначенный для создания программного обеспечения на основе взаимодействующих компонентов объекта, каждый из которых может использоваться во многих программах одновременно.

InfinityHMI – программное обеспечение, позволяющее визуализировать и управлять объектами технологического процесса на мнемосхемах в реальном масштабе времени производства компании АО ЭлеСи.

JSON - текстовый формат обмена данными, основанный на JavaScript.

ONVIF (Open Network Video Interface Forum) – отраслевая международная организация, которая занимается разработкой стандартизованных протоколов для взаимодействия различного оборудования и программных средств, входящих в состав систем безопасности.

OPC (Open Platform Communications) – семейство программных технологий (протоколов), предоставляющих единый интерфейс для управления объектами автоматизации и технологическими процессами.

PTZ (Pan, Tilt, Zoom) – аббревиатура от «панорамирования», «наклона» и «зума», которая описывает возможности управления камерой.

SCADA Infinity – программно-инструментальный комплекс для реализации автоматизированных систем управления технологическими процессами производства компании АО ЭлеСи.

SDK (Software Development Kit) – набор средств разработки, который позволяет специалистам по программному обеспечению создавать приложения для определённого пакета программ.

VMS (Video Management Service) – система управления видео.

Оглавление

Реферат	10
Перечень условных обозначений, единиц и терминов	11
Введение.....	15
1 ОБЗОР ПРЕДМЕТНОЙ ОБЛАСТИ	17
1.1 Основные понятия автоматизированных систем управления дорожным движением.....	17
1.1.1 Автоматические дорожные метеостанции	18
1.1.2 Детекторы транспортного потока.....	19
1.1.3 Видеонаблюдение	20
1.1.4 Знаки и табло переменной информации.....	21
1.1.5 Дорожные контроллеры	22
1.2 Обзор существующих аналогов.....	23
2 РАЗРАБОТКА ПОДСИСТЕМЫ ВИДЕОНАБЛЮДЕНИЯ	25
2.1 Постановка задач.....	25
2.2 Выбор сервера видеонаблюдения.....	27
2.3 Описание и структура подсистемы видеонаблюдения	29
Общий вывод по разделу.....	34
3 РАЗРАБОТКА ПОДСИСТЕМЫ МОНИТОРИНГА ТРАНСПОРТНЫХ ПОТОКОВ.....	35
3.1 Постановка задач.....	35
3.2 Разработка модели данных.....	36
3.3 Описание и структура подсистемы мониторинга транспортных потоков	38
Общий вывод по разделу.....	40
4 ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ.....	42
4.1 Оценка коммерческого потенциала, перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения.....	42
4.1.1 Анализ конкурентных решений.....	42
4.1.2 SWOT-анализ.....	44
4.2 Планирование и формирование бюджета научных исследований	45

4.2.1 Планирование этапов проведения работ	45
4.2.2 Бюджет научно-технического исследования	49
Общий вывод по разделу	56
5 СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ	57
5.1. Производственная безопасность	57
5.1.1. Параметры микроклимата	58
5.1.2. Освещенность рабочей зоны	60
5.1.3. Уровень шума	63
5.1.4 Умственное перенапряжение	64
5.1.5. Электробезопасность	65
5.2. Экологическая безопасность	67
5.3. Безопасность в чрезвычайных ситуациях	68
5.4 Правовые и организационные вопросы обеспечения безопасности	70
Общий вывод по разделу	71
Заключение	72
Список использованных источников	73
Приложение А	76
Приложение Б	78
Приложение В	84
Приложение Г	91

Введение

Наличие «умного» телевизора дома, мощного персонального компьютера, различных гаджетов уже никого не удивляет. Научно-технический прогресс шагнул далеко вперед и сложные системы, выполняющие огромное количество операций в секунду, принимающие сложные решения, заменяющие человека при выполнении рутинных и опасных работ, распространяются повсюду.

Довольно сложно представить сегодня организацию, предприятие или фирму в которых не задействованы автоматические или автоматизированные системы, управление которыми осуществляется при помощи компьютеризированных вычислительных систем, которые все больше заменяют человека или помогают ему в выполнении сложных задач, сокращая тем самым количество ошибок, связанных с человеческим фактором и улучшая общую производительность системы.

Контроль качества и правильности работы сложных механизмов, предотвращение противоправных действий, обнаружение опасных ситуаций (возгорание, проникновение и угроза личной или имущественной безопасности и др.) – вот далеко не полный перечень применения систем видеонаблюдения. К основным инфраструктурам, в которых обязательно применяются системы видеонаблюдения можно отнести следующие:

- аэропорты;
- банки;
- городское видеонаблюдение;
- общественный транспорт;
- морские порты и др.

Кроме систем видеонаблюдения существует также большое количество датчиков, сенсоров, радаров, которые помогают контролировать и фиксировать те параметры, которые недоступны для контроля системам видеонаблюдения.

Совместное использование таких устройств позволяет достичь наилучшего результата при мониторинге различных параметров.

Для автоматизации транспортной отрасли, в том числе снижение задержек транспорта, повышение безопасности дорожного движения, информирование участников дорожного движения, кроме самой системы видеонаблюдения необходимо использование различных радаров, детекторов, станций, информационных табло и различных систем связи.

Целью данной работы является разработка подсистем, позволяющих производить мониторинг дорожной ситуации и различных параметров транспортного потока, а также оперативное оповещение участников дорожного движения.

Объектом исследования является автоматизация дорожного движения.

Предметом исследования являются системы видеонаблюдения и системы учета интенсивности транспортного потока.

С практической точки зрения разрабатываемые подсистемы будут применяться в автоматизированной системе управления дорожным движением, основная часть которой разработана для использования в SCADA Infinity и InfinityHMI.

1 ОБЗОР ПРЕДМЕТНОЙ ОБЛАСТИ

1.1 Основные понятия автоматизированных систем управления дорожным движением

Перегруженность городских дорог, постоянные пробки создают не только множественные проблемы участникам дорожного движения, но и в целом городу или мегаполису. Концентрация выхлопных газов в местах скопления автомобилей увеличивается в несколько раз, также увеличивается и нагрузка на дорожное покрытие. Местные, не связанные и не согласованные между собой светофоры с морально устаревшим оборудованием, низкий уровень безопасности на дорогах, отсутствие связи со светофорами – это отсутствие оперативного контроля их работы и как следствие резкое повышение количества дорожно-транспортных происшествий. Отсутствие согласованной работы светофоров снижает общую пропускную способность и скорость транспортного потока.

Эта ситуация кардинально меняется при внедрении современных систем управления дорожным движением [4]. Системы управления можно разделить на две большие группы:

- системы управления городским движением;
- системы управления магистральным движением.

В данной работе будут рассматриваться только системы управления магистральным движением. Любая автоматизированная система получает информацию с датчиков, анализирует ее и генерирует управляющее воздействие. В связи с этим, все оборудование, используемое в АСУДД, может быть подразделено на две группы:

- оборудование собирающее информацию;
- оборудование оказывающее управляющее воздействие.

1.1.1 Автоматические дорожные метеостанции

Автоматические дорожные метеостанции (АДМС) - относятся к профессиональному оборудованию с цифровым интерфейсом для использования в метеорологических наблюдениях и прогнозах. К основным отличительным особенностям АДМС относятся:

- встроенная конструкция с вентилируемой радиационной защитой для проведения следующих измерений:

- относительная влажность;
- интенсивность осадков;
- количество осадков;
- тип осадков;
- температура воздуха;
- атмосферное давление;
- направление ветра
- скорость ветра.

- измерение относительной влажности производится с помощью чувствительного емкостного элемента;

- проведение измерения осадков осуществляется с помощью доплеровского радара, который производит измерения скорости каждой отдельной капли осадков (снег/дождь);

- расчет количества и интенсивности осадков достигается за счет корреляции между скоростью капли и ее размерами. Тип осадков (снег/дождь) обычно определяется на основе скорости падения капли

Также с метеостанцией обычно идет аппаратный шкаф, который оснащен контроллерами съема, производящими первичную обработку информации с метеодатчиков. Также в аппаратный шкаф включаются устройства связи, устройства защиты питания, устройства обогрева и ряд других.

1.1.2 Детекторы транспортного потока

Детектор транспортного потока предназначен для регистрации количества проходящих автомобильных средств через сечение дороги.

Кроме этого детектор транспорта определяет различные параметры транспортных потоков (среднее расстояние между транспортными средствами, средняя скорость движения, тип транспортного средства и др.).

Каждый детектор транспорта состоит из нескольких узлов:

- чувствительный элемент, который вырабатывает первичный сигнал во время прохождения автомобилем данного сечения дороги;
- усилительно-преобразовательный блок, необходимый для обработки первичного сигнала;
- выходной элемент, который передает закодированную информацию в другие устройства автоматической системы управления дорожного движения, например, контроллеры.

Все датчики классифицируются по нескольким параметрам: назначению, специализации, принципу работы чувствительного элемента.

По назначению детекторы транспорта разделяются на проходные и датчики присутствия.

Проходные датчики подают нормированные по продолжительности сигналы при появлении автомобиля в зоне, контролируемой им. Параметры выдаваемого сигнала от времени пребывания в этой зоне машины не зависят. То есть данное устройство способно лишь фиксировать факт появления транспортного средства. Это нужно для обеспечения алгоритма нахождения разрыва в потоке. По этой причине данные устройства нашли наибольшее распространение.

Датчики присутствия подают сигнал на протяжении всего времени пребывания транспортного средства в зоне, которая контролируется детектором транспорта. Данные датчики редко используются и предназначены в основном для определения начала образования затора.

По принципу работы чувствительного элемента детекторы транспортных потоков делятся:

- детекторы контактного типа:
 - электромеханические;
 - пневмоэлектрические;
 - пьезоэлектрические.
- детекторы излучения:
 - фотоэлектрические;
 - радарные;
 - ультразвуковые;
 - оптические.
- детекторы измерения электромагнитных систем:
 - ферромагнитные;
 - индуктивные.

Все перечисленные устройства имеют плюсы и минусы, и находят применение в той или иной ситуации.

1.1.3 Видеонаблюдение

Хотя сегодня автомагистрали оснащены множеством высокотехнологичных датчиков, которые великолепно отражают ситуацию на дорогах в любой момент времени, видеоизображение, при этом, гораздо важнее. Поэтому, системы видеонаблюдения являются наиболее широко распространёнными системами, которые расположенными на всех стратегических местах дорожной сети. Десятки и сотни видео потоков от систем дорожного видеонаблюдения постоянно собираются и обрабатываются в центрах управления дорожным движением.

Видеоинформацию с этих потоков необходимо обработать, чтобы сделать полезной для системы управления Инцидентами и Дорожным движением.

Системы автоматического видео обнаружения инцидентов и Видео анализа дорожного движения анализируют видео потоки с системы видеонаблюдения. Процесс обработки видео в режиме реального времени позволяет осуществить немедленное обнаружение дорожно-транспортных происшествий, заторов, нарушений и осуществить анализ транспортного потока, что является ключевым фактором в управлении инцидентами и дорожным движением.

Системы автоматического обнаружения инцидентов состоят из видеокамер, закрепленных над дорогой и правильно расположенных, а также из аналитического программного обеспечения, которое сообщает оператору о произошедшем инциденте и месте происшествия.

1.1.4 Знаки и табло переменной информации

Табло переменной информации (ТПИ) предназначено для визуального отображения информации обязательной для выполнения водителями или рекомендательной информации:

- информация о сбоях в движении (заторы, ДТП, остановки транспортных средств и т.д.);
- информация об ограничениях движения (при проведении специальных мероприятий правоохранительными органами, санкционированных массовых мероприятиях и т.д.);
- информация о проведении работ на проезжей части;
- информация о маршрутах объезда мест скопления транспорта; ограничении скорости движения (в том числе по метеорологическим причинам).

Знак переменной информации (ЗПИ) предназначен для выдачи водителям обязательной или рекомендательной информации об организации движения:

- указатель скорости;

- временное изменение организации движения при возникновении заторов, проведения массовых мероприятий, спец. проездов или для введения ограничений на проезд в определённое время суток;

- дублирование светофора на въезде на скоростную магистраль;

1.1.5 Дорожные контроллеры

Дорожный контроллер – это устройство, предназначенное для управления сигналами светофоров, расположенных на перекрестках. Они могут работать как самостоятельно, так и в составе автоматизированной системы управления движения.

Если установить дополнительную плату адаптивного управления, то дорожный контроллер может выполнять регулирование движения на основе информации, полученной от датчиков транспорта. При этом реализуются разнообразные алгоритмы динамического управления.

Контроллер имеет сложное строение, но фактически это компьютер в промышленном исполнении с дополнительными датчиками:

- радиомодем;
- модуль для подключения с IP-сети;
- модуль GPS;
- радиостанция и др.

Модульное использование дорожного контроллера облегчает техническое обслуживание, а также дает возможность довольно просто наращивать функциональные возможности, создавать необходимую архитектуру, заметно экономить денежные средства. После установки или замены любого блока работа возобновляется по заданной ранее схеме движения. При этом дополнительное программирование не нужно.

Дорожный контроллер устроен так, что к нему можно подключать различные внешние устройства, например, датчики движения, загазованности, информационного табло, ручные и автоматизированные пульта управления

движением. Затем он обрабатывает информацию, полученную от всех этих устройств.

Контроллеры, используемые локально, необходимы для регулировки и контроля движения на местных светофорных объектах. Если перегорает одна или большее количество красных ламп, то светофор автоматически переключится в режим желтого мигания.

Если возникает неправомерное включение зеленых ламп, то устройство обеспечивает отключение питания. При возникновении практически любой нештатной ситуации устройство срабатывает таким образом, что поломка ликвидируется.

1.2 Обзор существующих аналогов

При разработке программного продукта всегда необходимо принимать решение о целесообразности разработки, т.к. зачастую подобные задачи уже решались и приобретение готового решения оказывается в несколько раз выгоднее. Также не стоит забывать, что создание нового продукта требует больших временных затрат, а также наличие в штате компании специалистов необходимого уровня.

Все подсистемы разрабатываются для уже существующей SCADA Infinity, разработанной АО ЭлеСи, поэтому рассматривать аналогичные системы для автоматизации управления технологической линией нет необходимости в связи с тем, что данное программное обеспечение уже существует на рынке. Стоит также упомянуть, что данное программное обеспечение является узконаправленным и разрабатывалось специально для компании Транснефть, однако сейчас решения на базе SCADA Infinity применимы для предприятий, занимающимися различными отраслями.

На сегодняшний день существуют несколько компаний, предлагающих Вашему вниманию программно-аппаратные комплексы для автоматизации и управления дорожным движением.

Так компания «Технический центр по обеспечению безопасности дорожного движения» с 2014 года начала вводить в эксплуатацию АСУДД собственной разработки «Спектр». На сегодняшний день компанией внедрено восемьдесят один объект АСУДД в городе Пермь и четыре объекта в городе Тверь.

Следующая рассматриваемая система будет от группы компаний Индустрия света. Их автоматизированная система «КС» уже внедрена в нескольких городах России и Казахстана (Казань, Тамбов, Екатеринбург, Нижневартовск, Новоуральск, Караганда, Талдыкорган, Петропавловск, Кокшетау, Томск и ряде других городов).

Автоматизированная система управления дорожным движением «Агат» ОАО «АГАТ-Системы управления» Республики Беларусь также введена в эксплуатацию в ряде городов Республики Беларусь (Минск, Барановичи, Гродно), а также в городах Российской Федерации (Калининград, Новосибирск)

Также есть еще несколько систем, однако нет никакой информации об их внедрении, поэтому рассматриваться они не будут. Из рассмотренных выше систем трудно выделить приоритетную, т.к. все они обладают приблизительно одинаковой функциональностью и одинаковым аппаратным обеспечением.

Разработка АО ЭлеСи отличается от остальных тем, что данная система будет разработана для использования совместно со SCADA Infinity, что позволяет увеличить конкурентоспособность основного распространяемого программного продукта данной компании. Также разрабатываемая система может быть внедрена частично, например, для организации видеонаблюдения на предприятии.

В данной части были рассмотрены основные понятия автоматизированных систем управления дорожным движением, их аппаратную составляющую, краткие характеристики аппаратного обеспечения для АСУДД. Кроме того, рассмотрены существующие аналоги на рынке, функционально все системы имеют схожее строение и схожие принципы работы.

2 РАЗРАБОТКА ПОДСИСТЕМЫ ВИДЕОНАБЛЮДЕНИЯ

2.1 Постановка задач

Разрабатываемое программное обеспечение, в том числе и подсистема видеонаблюдения разрабатывается для использования совместно со SCADA Infinity и InfinityHMI – программное обеспечение, рассчитанное на оператора и позволяющее производить сбор информации, наблюдение за процессом, а также его управлением.

Основной используемой технологией является технология связывания и встраивания компонентов в другие компоненты COM, которая позволяет без особых сложностей внедрить подсистему в другую систему, поддерживающую данную технологию.

Наблюдение за обстановкой с использованием видеокамер, контроль и управление ими может производиться следующими способами:

- с помощью программного обеспечения, поставляемого совместно с камерой видеонаблюдения;
- с помощью специализированного программного обеспечения сторонних производителей.

Программное обеспечение, поставляемое совместно с камерой видеонаблюдения, имеет ряд недостатков:

- при использовании камер различных производителей используется различное ПО, интерфейсы, функционал. Это приводит к невозможности организации большого центра наблюдения и обработки данных;
- невозможность одновременного отслеживания нескольких видеокамер различных производителей;
- невозможность централизованного доступа к хранилищу видеоматериалов.

Программное обеспечение сторонних разработчиков обычно является более масштабируемым, более функциональным, но лучше заранее уточнить о

возможности поддержки оборудования. В связи с вышеперечисленными преимуществами и недостатками принято решение об использовании комплекта средств разработки (SDK) который предоставляется производителями профессионального программного обеспечения видеонаблюдения.

Разрабатываемая подсистема должна обеспечивать возможность выполнения перечисленных ниже функций:

- получения списка оборудования и присвоение ему определенного идентификационного номера, для возможности избирательного управления и размещения оборудования;
- оперативная передача информации от сервера видеонаблюдения в АСУДД;
- возможность просмотра условий движения из АСУДД в режиме реального времени;
- организация возможности удаленного управления камерами из АСУДД, обладающими соответствующей функциональностью;
- возможность создания алгоритма по последовательному переключению видеокамер;
- получение тревожных сообщений с камер, их трансляция в АСУДД;
- контроль наличия связи с сервером видеонаблюдения. В случае пропадания связи передача тревожного сигнала в АСУДД;
- выдача тревожного сигнала в случае пропадания связи с конкретной камерой;
- возможность получения доступа к архиву видеозаписей по запросу от оперативного персонала служб;
- возможность пометки видеофрагментов к долгосрочному хранению. С пометкой даты, времени и идентификатора камеры;
- возможность добавления комментариев к сохраняемым видеофрагментам или изображениям;

- возможность осуществления поиска по архиву видеозаписей по отдельным критериям (идентификатор видеокамеры, дата и время, комментарии);

- обеспечение возможности сохранения стоп-кадров в графический файл стандартного формата по требованию оператора или при фиксации определенного типа событий.

2.2 Выбор сервера видеонаблюдения

Axxon Next – система компании AxxonSoft может масштабироваться по заявлению разработчика бесконечно, не имея ограничений на число видеосерверов, рабочих станций или видеокамер. Данная система поддерживает более 6000 моделей IP устройств, включая более 1500 моделей IP видеокамер, использующих собственный интерфейс и 4500 ONVIF устройств. Одним из главных недостатков данной системы является то, что она не предоставляет API для интегрирования в собственные системы.

Zone Minder – открытая, кросс-платформенная система, поддерживающая множество USB и IP видеокамер. Настраиваемая система может использоваться для дома, систем малого бизнеса или корпоративных систем. Обладает совместимостью, начиная от Raspberry Pi, заканчивая самым последним серверным оборудованием. Преимуществами данной системы является открытость исходного кода, что позволяет лучше подстраивать ее для собственных нужд. Недостатком данной системы является

Xprotect – VMS-система (Video Management Service) на базе открытой платформы, предназначенная для крупных объектов, имеющих несколько филиалов и высокий уровень ответственности. Данная система обладает следующей функциональностью:

- поддержка неограниченного количества камер, пользователей и расположений.

- пользовательский интерфейс с централизованным управлением дает возможность контролировать все подключенные камеры, устройства, хранилища и пользователей.

- диспетчер Alarm Manager и интерактивные карты обеспечивают оперативную ситуационную осведомленность, помогая операторам оптимизировать оценку происшествий и быстро принимать требуемые меры.

- возможность централизации управления несколькими системами, при помощи Milestone Federated Architecture™.

- отличные показатели отказоустойчивости благодаря автоматическому резервированию серверов функции Edge Storage, обеспечивают непрерывность записи видео и постоянный доступ к системе.

Программное обеспечение для управления видео (Video Management Service) на базе открытой платформы XProtect от Milestone интегрируется с наиболее широким в отрасли рядом видеокамер, предоставляя свободу выбора оборудования в соответствии с конкретными потребностями и бюджетом. Стремясь предоставить действительно открытую платформу для IP-видео и обеспечить свободу выбора, компания Milestone гордо занимает место в рядах поставщиков первых VMS-систем, поддерживающих и развивающих стандарты ONVIF и PSIA. Простота интеграции приложения в VMS XProtect при помощи пакета разработки программ интеграционной платформы Milestone (MIP SDK) делает ее одной из самых популярных VMS-систем. Открытая архитектура XProtect в сочетании с MIP SDK открывает безграничные возможности по добавлению функций и функциональности для создания мощных пользовательских решений для видеонаблюдения.

Проведя анализ различных систем видеонаблюдения, разрабатываемая подсистема будет реализовываться с использованием комплекта средств разработки (SDK) предоставленным производителями профессионального программного обеспечения видеонаблюдения Milestone.

2.3 Описание и структура подсистемы видеонаблюдения

Краткое наименование – подсистема AM.Eye.

Подсистема AM.Eye состоит из двух компонентов:

- мультикамерный компонент AM.Eye-Big;
- однокамерный компонент AM.Eye-Mini.

Однокамерный компонент AM.Eye-Mini позволяет получить доступ к просмотру видеопотока с одной конкретной камеры, которая встраивается непосредственно в мнемосхему, на которой, помимо этого компонента, расположены другие элементы управления и мониторинга интеллектуальной транспортной системы.

Архитектура программного обеспечения [Рисунки 1,2] разрабатывалась с учетом того, что сервер видеонаблюдения в дальнейшем может измениться.

AM.Eye – Mini конструктивно состоит из 4 модулей:

- модуль наблюдения и управления;
- диспетчер конфигураций;
- модуль наблюдения;
- модуль авторизации.

Каждый из этих модулей использует ту или иную часть Milestone SDK для взаимодействия с различными частями Milestone XProtect.

Используемые модули SDK логически могут быть разделены на три части:

- Image Viewer ActiveX;
- Audio Player ActiveX;
- MIP .Net Library;

Эти части отвечают за управление видеопотоком, аудиопотоком и работу с данными, получаемыми с серверов, соответственно. Все взаимодействия с осуществляются по протоколу TCP/IP.

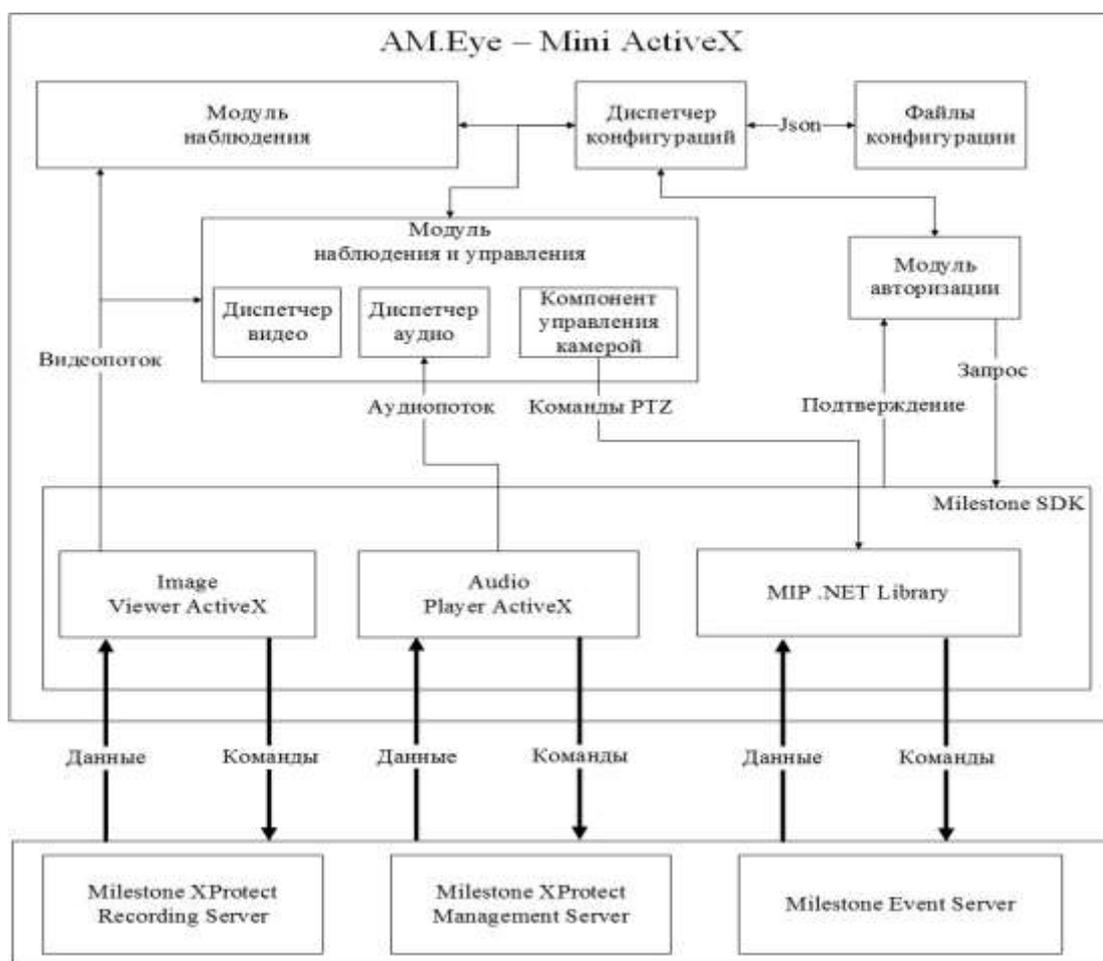


Рисунок 1 – Архитектура компонента подсистемы видеонаблюдения
AM.Eye – Mini

Модуль авторизации используется для распознавания идентификационных данных пользователя на сервере Milestone. После введения данных пользователя и данных сервера, этому серверу отправляется запрос с использованием средств MIP .Net Library и, в случае успешной авторизации, компонент переходит в режим получения видео, аудио и данных от видеосервера. Для подтверждения или отказа в запросе авторизации используется информация Milestone Management Server, который должен быть предварительно настроен и иметь запрашиваемого пользователя с соответствующими привилегиями.

При успешном входе и сохранении мнемосхемы данные пользователя, сервера и выбранная видеочамера сохраняются диспетчером конфигураций в

файлы конфигурации в формате JSON, откуда затем при каждом запуске мнемосхемы подгружаются в автоматическом режиме.

Каждый AM.Eye – Mini, расположенный в приложении визуализации, работает как полноценный независимый компонент, однако, если несколько компонентов подключены к одному и тому же видеосерверу все они начинают работать в режиме использования общих компонентов, т.е. если одним компонентом выполнена авторизация на сервере, то остальные компоненты используют данную информацию что повышает стабильность работы системы, а также приводит к снижению нагрузки на видеосервер и количеству переданных данных.

Диспетчер конфигураций отвечает за сохранения пользовательских настроек (имён камер, заданных пользователем, данных авторизации и т.д.) в формате JSON в файлах конфигурации и получение этих данных при необходимости. Диспетчер предполагает использование и изменение настроек только внутри приложения, поэтому внешнее изменение файлов конфигурации нежелательно и может привести к сбоям в работе программы.

Модуль наблюдения и управления отвечает за получения видео и аудио информации с камер, а также за управление камерами, поддерживающими протокол PTZ по протоколу IP. Кроме того, модуль выполняет роль контейнера для нескольких Image Viewer ActiveX и Audio Player ActiveX, облегчая пользователю работу одновременно с несколькими камерами, а также предоставляет удобный интерфейс для отправки команд серверам Milestone. Модуль использует Image Viewer ActiveX для получения видео в виде потока байт с сервера, который затем преобразуется компонентом Image Viewer в видео без участия AM.Eye – Mini. Аналогичным образом получается аудио. Image Viewer и Audio Player после успешной авторизации на сервере работают в автоматическом режиме, без необходимости ручного переподключения или дополнительной настройки. Однако, на случай непредвиденных ситуаций, существует возможность ручного запроса восстановления соединения с сервером.

Управление камерой (PTZ) осуществляется за счет отправки серверу соответствующих команд с использованием SDK (MIP .Net Library). Настройки камер и групп камер (имена, Id, описание) сохраняются диспетчером конфигураций в файлы JSON.

Мультикамерный компонент является полнофункциональным приложением, с возможностью работы с несколькими камерами одновременно.

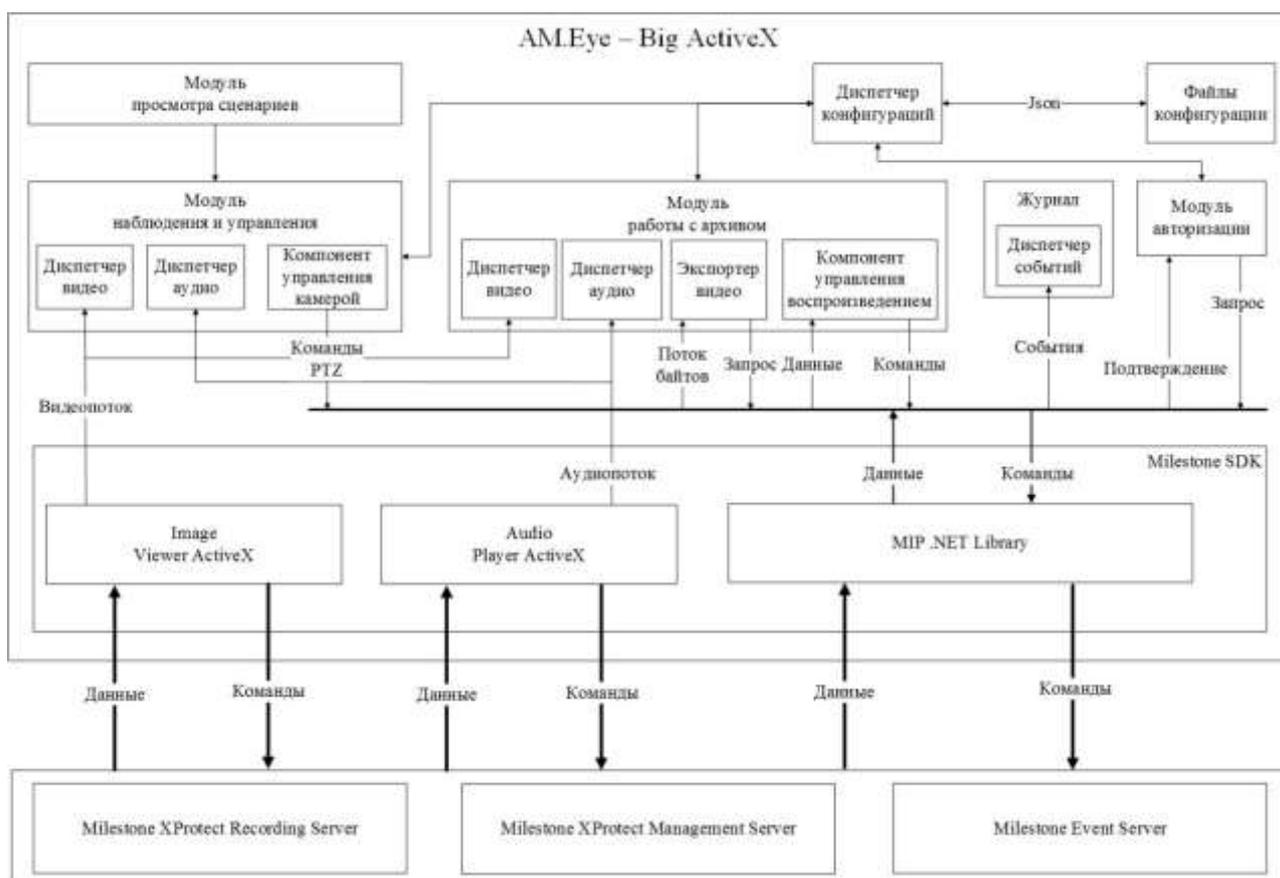


Рисунок 2 – Архитектура компонента подсистемы видеонаблюдения AM.Eye – Big

AM.Eye – Big конструктивно состоит из 6 модулей:

- модуль просмотра сценариев;
- модуль наблюдения и управления;
- модуль работы с архивом;
- журнал;
- диспетчер конфигураций;

- модуль авторизации.

Каждый из этих модулей использует ту или иную часть Milestone SDK, также, как и компонент подсистемы AM.Eye – Mini.

Модуль авторизации, диспетчер конфигураций, модуль наблюдения и управления описаны выше при описании компонента AM.Eye – Mini подсистемы видеонаблюдения. Принцип их работы и функциональность являются схожими (более того, для сокращения времени разработки оба компонента используют повторно один модуль просмотра и наблюдения), поэтому описываться повторно не будут.

Журнал автоматически получает события с Milestone Event Server при помощи встроенного протокола, предоставленного компанией и поддерживает добавление записей из внутренней структуры приложения, например, для тех случаев, когда отследить событие не представляется возможным (Потеря соединения с сервером). Все записи журнала сохраняются в файл формата JSON, максимальный размер которого задаётся пользователем и варьируется в интервале от 10 до 1000 Мб.

Модуль просмотра сценариев использует уже настроенные в модуле наблюдения и управления компоненты для зацикленного поочерёдного отображения видео с нескольких камер в порядке и с интервалами, заданными пользователем. При таком использовании модуля наблюдения и управления теряется возможность использовать компонент управления камерой.

Модуль работы с архивом осуществляет получение видео и аудио информации тем же способом, что и модуль наблюдения и управления, однако момент времени, для которого необходимо получить аудио и видео устанавливается при помощи запроса к Milestone Recording Server, который после получения запроса начинает посылать пакеты информации о выбранном моменте времени. Кроме того, компонент управления воспроизведением может ускорять/замедлять и останавливать видеопоток, а также переходить между последовательностями (цельный отрезок видео, длиной не более 1 часа) и закливать выбранные участки видео. Для более удобного использования, в

компоненте реализован поиск по архиву существующих на сервере видеозаписей средствами SDK (MIP .Net Library).

Компоненты наблюдения и управления, просмотра сценариев и работы с архивом получают аудио и видео информацию с Milestone Recording Server, следовательно, его стабильная работа обязательна для работы приложения.

Внутри модуля работы с архивом существует компонент экспорта видео, который способен, используя MIP .Net Library, запрашивать у Milestone Recording Server выбранные для экспорта фрагменты видео, которые тот передаёт в виде байтового потока, а компонент экспорта сохраняет в выбранном пользователем формате. На данный момент поддерживаются форматы:

- AVI (кодек может быть выбран из списка системных кодеков);
- MKV (встроенный кодек);
- BLK (формат Milestone).

Общий вывод по разделу

В ходе выполнения работы разработана подсистема, реализующая следующую функциональность:

- выполнение компонент в качестве COM компонентов с возможностью встраивания в приложение визуализации процесса, являющееся для него контейнером или любое другое поддерживающее данную технологию;
- просмотр видеопотока с нескольких видеокамер одновременно;
- ручное управление видеокамерами;
- автоматическое переключение между камерами и группами камер (создание и использование сценариев переключения групп и камер);
- архивирование видеоданных и предоставление возможности их последующей обработки;
- ведение журналов событий, тревог и аварий;
- контроль состояния периферийного оборудования и выдача тревожных сообщений.

3 РАЗРАБОТКА ПОДСИСТЕМЫ МОНИТОРИНГА ТРАНСПОРТНЫХ ПОТОКОВ

3.1 Постановка задач

Разрабатываемое программное обеспечение предназначено для обеспечения сбора в реальном времени информации от датчиков потока Smartmicro UMRR TYPE 39 о загруженности автодороги и ее передачу в автоматизированную систему управления дорожного движения.

Целью разработки данной подсистемы является агрегирование данных о дорожном потоке, на их основании реализация функций управления дорожным движением, улучшение уровня информирования участников дорожного движения, прогнозирование нагрузки на транспортные узлы в зависимости от внешних факторов.

Подсистема должна обеспечивать решение следующих задач:

- распознавание оборудования и присвоение ему определенного идентификационного номера, для возможности избирательного управления и размещения оборудования;

- оперативная передача информации от датчиков потока в автоматизированную систему управления дорожного движения. С возможностью задания интервала передачи собранных данных.

Получение данных транспортного потока:

- a) автоматического обнаружения заторов;
- b) остановившихся транспортных средств;
- c) транспортных средств, движущихся навстречу направлению движения;
- d) количество проехавших транспортных средств;
- e) скорость транспортных средств;
- f) временной интервал между транспортными средствами и другая информация от датчиков транспортного потока.

- возможность определения процента загрузки по полосам;
- возможность определения конца очереди из транспортных средств;
- возможность задания интервала времени, за который производится анализ предоставляемой информации;
- возможность задания интервала опроса периферийного оборудования, по истечению которого, если ответ от датчика потока не получен, констатировать факт отсутствия связи с датчиком потока и присваивать последним данным с него плохого качества.
- передача в АСУДД данных о неисправном оборудовании с его идентификационным номером.
- автоматическое восстановление связи с исправным оборудованием;
- возможность включения, отключения, настройки записи и просмотра лог-файлов, содержащих обмен данными между модулем и периферийным оборудованием.
- работа с остальными элементами АСУДД посредством OPC технологии.

3.2 Разработка модели данных

В ходе изучения технической документации по датчикам Smartmicro UMRR-0A, UMRR-0C Type 39, UMRR-0C Type 40, UMRR-0C Type 42, поступило предложение объединить усилия совместно с компанией «СОРБ Групп» основные направления деятельности которой:

- разработка и внедрение автоматизированных систем поддержки существующих бизнес-процессов предприятия;
- системная интеграция;
- организация и поддержка центров обработки данных и ряд других.

Данная компания уже вела разработку универсального протокола для поддержки всех перечисленных выше типов датчиков на протяжении полугода. В связи с этим принято решение в объединении усилий с «СОРБ Групп» т.к. в этом есть ряд преимуществ:

- сокращение времени разработки на разбор протоколов для каждого типа датчика потока;

- поддержка сразу нескольких типов датчиков потока;

- гарантированная доставка сообщений от датчиков потока;

В ходе дальнейшего взаимодействия с компанией совместно был разработан формат сообщений с датчиков потока для статистических данных, а также для данных инцидентов:

```
{
  "time_start": "YYYY-MM-DD HH:MM:SS",
  "time_end": "YYYY-MM-DD HH:MM:SS",
  "sensors": [{
    "id": 0,
    "code": "Some Road",
    "date_from": "YYYY-MM-DD HH:MM:SS",
    "sensor_status_id": true,
    "place_name": "M123 (Some region)",
    "latitude": "250.04",
    "longitude": "150.23",
    "parent_id": 18,
    "lanes": [{
      "lane_id": 0,
      "class0": 1,
      "class1": 0,
      "class2": 5,
      "class3": 6,
      "class4": 9,
      "occupancy": "9.00",
      "av_speed": "84",
      "speed85": "97",
      "gap": "91",
      "total_num": 21
    },
    ...
  ]
},
...
]
```

3.3 Описание и структура подсистемы мониторинга транспортных потоков

Подсистема обеспечивает сбор в реальном времени информации от датчиков дорожного потока о загруженности автодороги и её передачу в информационную транспортную систему для последующей обработки и анализа.

Назначение подсистемы - агрегировать данные о дорожном потоке, на их основании реализовывать функции управления дорожным движением, улучшать уровень информирования участников дорожного движения, осуществлять прогнозирование нагрузки на транспортные узлы в зависимости от внешних факторов, разрабатывать превентивные меры по увеличению мобильности населения.

Подсистема обеспечивает получение данных от датчиков мониторинга трафика компании Smartmicro.

Структурой подсистемы [Рисунок 3] представлена возможность работы с различными типами дорожных датчиков, работающих в соответствии с стандартизированной классификацией транспортных средств.

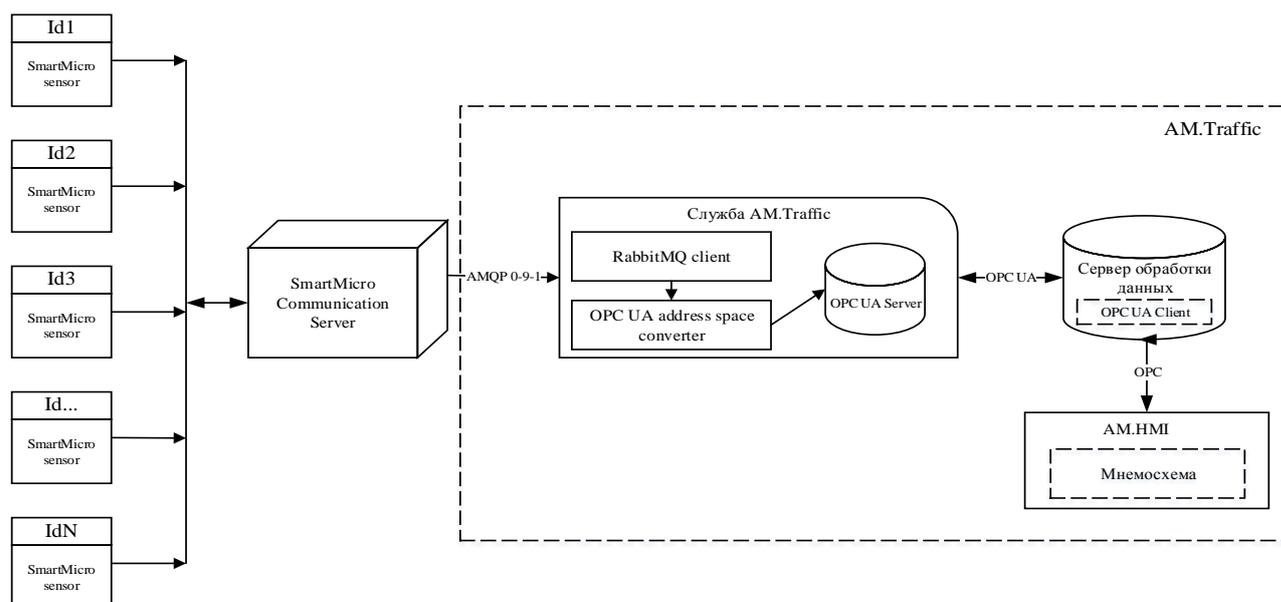


Рисунок 3 – Архитектура подсистемы мониторинга характеристик транспортных потоков

Краткое наименование – подсистема AM.Traffic.

Структурная схема системы [Рисунок 3] обеспечивает гарантированную доставку сообщений о дорожной ситуации. Гарантированность доставки реализуется за счет использования протокола AMQP 0-9-1 (Advance Message Queuing Protocol) - открытый протокол для передачи сообщений между компонентами системы. Основная идея состоит в том, что отдельные подсистемы (или независимые приложения) могут обмениваться произвольным образом через AMQP-брокер, который осуществляет маршрутизацию, гарантирует доставку, распределение потоков данных, подписку на нужные типы сообщений.

Сервер, выполняющий непосредственное взаимодействие с датчиками выполняет роль высокоуровневого интерфейса над оборудованием. Данный сервер собирает статистическую информацию (с определенным интервалом времени) с датчиков и помещает в базу данных, заполняя, при этом очередь сообщений, к которой подключается служба при помощи RabbitMQ (платформа, реализующая систему обмена сообщениями между компонентами программной системы на основе стандарта AMQP) клиента. В случае если служба была отключена от сервера очередь сообщений пополняется и при подключении последовательно из очереди забираются все сообщения. Когда сообщений в очереди нет, служба находится в режиме ожидания. При получении нового сообщения, сервер оповещает службу о нем и передает ей это сообщение для обработки.

При получении сообщения с информацией с датчиков [Пункт 3.2], служба при помощи конвертера формирует адресное пространство OPC UA, после чего информация отправляется на сервер OPC UA к которому подключается в качестве клиента сервер обмена данными. От сервера обмена данными информация по внутреннему протоколу OPC поступает в приложение визуализации для дальнейшего предоставления актуальной информации о качественном и количественном состоянии транспортных потоков на контролируемом сечении автодороги.

Общий вывод по разделу

В ходе выполнения работы разработана подсистема, обеспечивающая решение следующих задач:

- распознавание оборудования и присвоение ему определённого идентификационного номера, для возможности избирательного управления и контроля;

- оперативная передача информации от датчиков потока в информационную транспортную систему. С возможностью задания интервала передачи агрегированных данных;

- получение данных транспортного потока:

- a) автоматическое обнаружение заторов,

- b) обнаружение остановившихся транспортных средств;

- c) обнаружение транспортных средств, движущихся навстречу направлению движения;

- d) подсчет количества проехавших транспортных средств;

- e) подсчет скорости проехавших транспортных средств, скорость;

- f) подсчет временного интервала между транспортными средствами;

- возможность определения качественных характеристик транспортного потока:

- a) процента загрузки по полосам;

- b) определение конца очереди из ТС.

- задание интервала времени, за который производится получение объёма информации;

- установка интервала опроса периферийного оборудования. По истечению, которого, если ответ не получен, констатация факта отсутствия связи с датчиком и присвоение последним данным с него плохого качества;

- передача в единую платформу данных о неисправном оборудовании с конкретизацией его идентификационного номера;

- автоматическое восстановление связи с исправным оборудованием;
- возможность включения/отключения/настройки записи и просмотра лог-файлов, содержащих обмен данными между модулем и периферийным оборудованием;
- работа с остальными элементами информационной транспортной системы посредством OPC UA технологии.

Компонент AM.Traffic создан для использования как в узкоспециализированных подсистемах для осуществления анализа и ведения статистики контролируемых транспортных потоков так и в составе интеллектуальных транспортных систем для оперативного осуществления доступа к дорожной информации в режиме реального времени из единой интеграционной платформы и выработки ответной реакции системы на транслируемые события, обеспечения тем самым оперативного реагирования всеми компонентами и поддержания высокого уровня безопасности и адаптивности.

4 ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ

Целью данного раздела является комплексное описание и анализ финансово-экономических аспектов работы. Необходимо оценить бюджет проекта, а также дать приближенную экономическую оценку результатов его внедрения, что позволит с помощью показателей эффективности оценить экономическую целесообразность осуществления работы.

4.1 Оценка коммерческого потенциала, перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения

4.1.1 Анализ конкурентных решений

Детальный анализ конкурирующих разработок, существующих на рынке, необходимо проводить систематически, поскольку рынки пребывают в постоянном движении. Такой анализ помогает вносить коррективы в научное исследование, чтобы успешнее противостоять своим соперникам. Делая упор на слабые места конкурентов можно получить большое преимущество на рынке. Важно реалистично оценить сильные и слабые стороны разработок конкурентов.

С этой целью может быть использована вся имеющаяся информация о конкурентных разработках:

- технические характеристики разработки;
- конкурентоспособность разработки;
- уровень завершенности научного исследования;
- бюджет разработки
- уровень проникновения на рынок;
- финансовое положение конкурентов, тенденции его изменения и т.д.

Анализ конкурентных технических решений с позиции ресурсоэффективности и ресурсосбережения позволяет провести оценку сравнительной эффективности научной разработки и определить направления для ее будущего повышения.

Наиболее конкурентные автоматизированные системы управления дорожным движением перечислены ниже. Оценочная карта для сравнения конкурентных решений представлена в таблице 1.

1. АСУДД «Агат».
2. АСУДД «КС».
3. АСУДД «Спектр».

Таблица 1 – Оценочная карта сравнения конкурентных решений

<i>Критерии оценки</i>	<i>Вес критерия</i>	<i>Баллы</i>				<i>Конкурентоспособность</i>			
		<i>Б_ф</i>	<i>Б₁</i>	<i>Б₂</i>	<i>Б₃</i>	<i>К_ф</i>	<i>К₁</i>	<i>К₂</i>	<i>К₃</i>
1	2	3	4	5	6	7	8	9	10
<i>Критерии оценки ресурсоэффективности</i>									
1. Функциональная мощность	0,25	4	4	4	4	1	1	1	1
2. Удобство в эксплуатации	0,15	4	5	4	4	0,6	0,75	0,6	0,6
3. Доступность	0,1	4	4	3	4	0,4	0,4	0,3	0,4
4. Качество интерфейса	0,15	5	5	4	4	0,75	0,75	0,6	0,6
<i>Экономические критерии оценки эффективности</i>									
1. Конкурентоспособность продукта	0,1	4	4	3	4	0,4	0,4	0,3	0,4
2. Цена	0,05	4	2	5	4	0,2	0,1	0,25	0,2
3. Поддержка продукта	0,1	5	5	4	4	0,5	0,5	0,4	0,4
4. Уровень проникновения на рынок	0,1	4	3	5	4	0,4	0,3	0,5	0,4
Итого	1	34	32	32	32	4,25	4,2	3,95	4,0

В соответствии с проведенными расчетами можно прийти к выводу, что все автоматизированные системы в среднем находятся в одинаковых условиях конкуренции, обладая при этом своими преимуществами и недостатками.

4.1.2 SWOT-анализ

Для выявления сильных и слабых сторон проекта необходимо провести SWOT-анализ на основе анализа рынка и анализа конкурентных решений. Результаты проведения анализа сводятся в матрицу SWOT [Таблица 2].

На втором этапе составляются интерактивные матрицы проекта, в которых отражено соответствие параметров SWOT [Таблица 3].

Таблица 2 – Матрица SWOT

	<p><i>Сильные стороны проекта:</i> С1. Удобство эксплуатации С2. Легкость настройки С3. Функциональная мощность (предоставляемые возможности) С4. Модульность системы. С5. Географическая распределенность системы</p>	<p><i>Слабые стороны проекта:</i> Сл1. Значительные временные и интеллектуальные затраты на реализацию. Сл2. Зависимость от других программных продуктов. Сл3. Невозможность оперативной поддержки от сторонних разработчиков. Сл4. Высокие требования к ресурсам.</p>
<p><i>Возможности:</i> В1. Ранний выход на рынок. В2. Появление спроса на разрабатываемую систему. В3. Использование финансирования научной деятельности В4. Привлечение специалистов компании ЭлеСи для работы над проектом. В5. Участие в различных выставках и форумах.</p>	<p>Функциональная мощность достигается за счет дополнительного финансирования, и привлечения сторонних специалистов. Модульность системы позволяет выпустить различные части по отдельности что приводит к дополнительному спросу и раннему выходу на рынок</p>	<p>Зависимость от других программных продуктов способствует более раннему выходу на рынок. Привлечение дополнительных специалистов приводит к значительным интеллектуальным затратам</p>
<p><i>Угрозы:</i> У1. Появление более качественных аналогов У2. Окончание поддержки видеосервера У3. Окончание поддержки API компанией СОРБ У4. Сложность разрабатываемой системы. У5. Появление более сильного конкурента.</p>	<p>Удобство эксплуатации, легкость настройки, модульность системы, функциональная мощность и географическая распределенность сильно усложняют систему.</p>	<p>Невозможность оперативной поддержки сторонними разработчиками программного обеспечения приводит к усложнению системы. Также зависимости от стороннего программного обеспечения могут привести к замене старого и дополнительным интеллектуальным и временным затратам</p>

Таблица 3 – Интерактивная матрица проекта

		Сильные стороны проекта					Слабые стороны проекта			
		C1	C2	C3	C4	C5	Сл1	Сл2	Сл3	Сл4
Возможности проекта	B1	-	-	-	+	-	-	+	-	+
	B2	+	+	+	+	+	0	-	-	-
	B3	-	-	+	-	0	+	-	+	-
	B4	-	-	+	-	-	+	+	+	-
	B5	-	-	0	-	-	-	-	-	-
Угрозы проекта	У1	-	-	0	0	+	+	+	0	+
	У2	-	-	0	-	-	0	+	+	-
	У3	-	-	0	-	-	0	+	+	-
	У4	+	+	+	+	+	+	-	+	0
	У5	-	-	+	+	+	+	0	-	0

Положительные и слабые стороны проекта, которые были выделены в ходе проведенного анализа, дают возможность спланировать необходимые изменения, слабые стороны проекта необходимо по возможности минимизировать, опираясь прежде всего на имеющиеся сильные стороны.

4.2 Планирование и формирование бюджета научных исследований

4.2.1 Планирование этапов проведения работ

Планирование комплекса предполагаемых работ осуществляется в следующем порядке:

- определение структуры работ в рамках научного исследования;
- определение участников каждой работы;
- установление продолжительности работ;
- построение графика проведения научных исследований.

Для выполнения научных исследований формируется рабочая группа, в состав которой могут входить научные сотрудники и преподаватели, инженеры, техники и лаборанты, численность групп может варьироваться. По каждому виду запланированных работ устанавливается соответствующая должность исполнителей.

В разработке системы задействованы следующие исполнители:

- начальник отдела разработки транспортных систем АО ЭлеСи (начальник ОРТС);
- научный руководитель;
- программист (П1);
- программист (П2);
- программист (П3);
- бакалавр (Б).

Таблица 4 – Перечень этапов, работ и распределение исполнителей

<i>Основные этапы</i>	<i>№ раб</i>	<i>Содержание работ</i>	<i>Должность исполнителя</i>
Разработка технического задания	1	Составление и утверждение технического задания	Начальник ОРТС
Выбор направления исследований	2	Подбор материалов по теме	П1, П2, Б
	3	Изучение материалов по теме	П1, П2, П3, Б
	4	Выбор направления	П1, Начальник ОРТС
	5	Календарное планирование работ по теме	Начальник ОРТС
Проектирование структуры и разработка подсистем	6	Проектирование структуры подсистемы видеонаблюдения	П1, Б
	7	Проектирование структуры подсистемы мониторинга	Б
	8	Разработка подсистемы видеонаблюдения	П1, П2, П3, Б
	9	Разработка подсистемы мониторинга	Б
	10	Тестирование подсистемы видеонаблюдения	П1, П2, Б
	11	Внесение изменений в подсистему видеонаблюдения	Б
	12	Тестирование подсистемы мониторинга	Б
Обобщение и оценка результатов	13	Оценка эффективности полученных результатов	Начальник ОРТС, НР

Основные этапы	№ раб	Содержание работ	Должность исполнителя
Оформление отчета по НИР	14	Составление отчета НИОКР	Начальник ОРТС, ПЗ
	15	Составление пояснительной записки	Б, НР

Расчет продолжительности этапов работ при выполнении выпускной квалификационной работы является важным этапом, так как трудовые затраты составляют основную часть стоимости научно-исследовательской работы (НИР).

Существуют разные методы расчета продолжительности этапов работы, в рамках данной работы используется экспертный способ. Он предполагает генерацию необходимых количественных оценок специалистами конкретной предметной области, опирающимися на их профессиональный опыт и эрудицию.

Для определения вероятных (ожидаемых) значений продолжительности работ $t_{ож}$ применяется следующая формула:

$$t_{ож i} = \frac{3t_{min i} + 2t_{max i}}{5} \quad (1)$$

где $t_{ож i}$ – ожидаемая трудоемкость выполнения i -ой работы чел.-дн.;

$t_{min i}$ – минимально возможная трудоемкость выполнения заданной i -ой работы, чел.-дн.;

$t_{max i}$ – максимальная возможная трудоемкость выполнения заданной i -ой работы, чел.-дн.

Для выполнения перечисленных работ [Таблица 4] требуются специалисты:

- начальник ОРТС;
- научный руководитель;
- программист (4 человека).

Исходя из ожидаемой трудоемкости работ, определяется продолжительность каждой работы в рабочих днях T_p , учитывающая параллельность выполнения работ несколькими исполнителями. Такое вычисление необходимо для обоснованного расчета заработной платы, так как

удельный вес зарплаты в общей сметной стоимости научных исследований составляет около 65 %.

$$T_{pi} = \frac{t_{ожi}}{Ч_i} \quad (2)$$

где T_{pi} – продолжительность одной работы, раб. дн.;

$t_{ожi}$ – ожидаемая трудоемкость выполнения одной работы, чел.-дн.;

$Ч_i$ – численность исполнителей, выполняющих одновременно одну и ту же работу на данном этапе, чел.

Для удобства построения графика, длительность каждого из этапов работ из рабочих дней следует перевести в календарные дни. Для этого необходимо воспользоваться следующей формулой:

$$T_{ki} = T_{pi} \times k_{\text{кал}} \quad (3)$$

где T_{ki} – продолжительность выполнения i -й работы в календарных днях;

T_{pi} – продолжительность выполнения i -й работы в рабочих днях;

$k_{\text{кал}}$ – коэффициент календарности.

Коэффициент календарности определяется по следующей формуле:

$$k_{\text{кал}} = \frac{T_{\text{кал}}}{T_{\text{кал}} - T_{\text{вых}} - T_{\text{пр}}} \quad (4)$$

где $T_{\text{кал}}$ – календарные дни ($T_{\text{кал}} = 365$);

$T_{\text{вд}}$ – выходные дни ($T_{\text{вд}} = 90$);

$T_{\text{пр}}$ – праздничные дни ($T_{\text{пр}} = 28$).

$$k_{\text{кал}} = \frac{365}{365 - 90 - 28} = 1,48 \quad (5)$$

Временные показатели проведения различных этапов работ отражены в таблице А1.

Для наглядного отображения графика и распределения работ между участниками проекта использована диаграмма Ганта. Диаграмма Ганта

представляет собой ленточный график, на котором работы по теме представляются протяженными во времени отрезками, характеризующиеся датами начала и окончания выполнения того или иного этапа работ.

Для построения диаграммы календарных работ используются данные, рассчитанные ранее и представленные в таблице А1.

Сам график работ в виде диаграммы Ганта приведен в таблице А2.

4.2.2 Бюджет научно-технического исследования

В состав затрат на разработку ПО и проведение эксперимента включается стоимость всех расходов, необходимых для реализации комплекса работ, составляющих содержание данной разработки. Расчет сметной стоимости на выполнение данной разработки производится по следующим статьям затрат:

- материальные затраты;
- затраты на специальное оборудование;
- основная заработная плата;
- дополнительная заработная плата;
- отчисления во внебюджетные фонды;
- контрагентные расходы;
- накладные расходы.

4.2.2.1 Материальные затраты

Произведем расчет всех материалов, используемых при разработке проекта:

- приобретаемые со стороны сырье и материалы, необходимые для создания научно-технической продукции;
- покупные материалы, используемые в процессе создания научно-технической продукции для обеспечения нормального технологического процесса и для упаковки продукции или расходуемых на другие

производственные и хозяйственные нужды (проведение испытаний, контроль, содержание, ремонт и эксплуатация оборудования, зданий, сооружений, других основных средств и прочее), а также запасные части для ремонта оборудования, износа инструментов, приспособлений, инвентаря, приборов, лабораторного оборудования и других средств труда, не относимых к основным средствам, износ спецодежды и других малоценных и быстроизнашивающихся предметов;

- покупные комплектующие изделия и полуфабрикаты, подвергающиеся в дальнейшем монтажу или дополнительной обработке;

- сырье и материалы, покупные комплектующие изделия и полуфабрикаты, используемые в качестве объектов исследований (испытаний) и для эксплуатации, технического обслуживания и ремонта изделий – объектов испытаний (исследований);

В материальные затраты, помимо вышеуказанных, включаются дополнительно затраты на канцелярские принадлежности, диски, картриджи и т.п. Однако их учет ведется в данной статье только в том случае, если в научной организации их не включают в расходы на использование оборудования или накладные расходы. В первом случае на них определяются соответствующие нормы расхода от установленной базы. Во втором случае их величина учитывается как некая доля в коэффициенте накладных расходов.

Расчет материальных затрат осуществляется по следующей формуле:

$$Z_m = (1 + k_T) \times \sum_{i=1}^m C_i \times N_{\text{расх } i} \quad (6)$$

где m – количество видов материальных ресурсов, потребляемых при выполнении научного исследования;

$N_{\text{расх } i}$ – количество материальных ресурсов i -го вида, планируемых к использованию при выполнении научного исследования (шт., кг, м, и т.д.);

C_i – цена приобретения единицы i -го вида потребляемых материальных ресурсов (руб./шт., руб./кг, руб./м и т.д.);

k_T – коэффициент учитывающий транспортно-заготовительные расходы.

Материальные затраты сведены в таблицу 5.

Таблица 5 – Материальные затраты

Наим.	Ед. изм.	Количество					Цена за ед., руб.					Затраты на материалы, руб.				
		ОРГС	Прогр.1	Прогр.2	Прогр.3	Бакалавр	ОРГС	Прогр.1	Прогр.2	Прогр.3	Бакалавр	ОРГС	Прогр.1	Прогр.2	Прогр.3	Бакалавр
<i>Офисные принадлежности</i>																
Бумага для принтера А4	уп	3	0,25		0,15	0,15	260	260		260	39	780	65		39	39
Картридж для принтера	шт	1					1500					1500				
Папка со скоросшивателем	шт	5			1		70			70		350			70	
<i>Затраты на электроэнергию</i>																
Электроэнергия	кВт	276	733	670	747	1245	3,25	3,25	3,25	3,25	3,25	897	2382,25	2177,5	2427,75	4046,25
<i>Итого</i>												2177	2447,25	2177,5	2536,75	4085,25

4.2.2.2 Расчет затрат на специальное оборудование

В данной статье расходов учитываются затраты на приобретение специального оборудования, необходимого для проведения работ по разрабатываемой теме.

В данном случае приобреталась только скоростная IP видеочкамера фирмы Dahua в количестве 1 шт., стоимостью 65000 рублей.

4.2.2.3 Основная заработная плата исполнителей

В данном разделе рассчитывается основная заработная плата работников, непосредственно занятых выполнением работы, (включая премии, доплаты) и дополнительную заработную плату:

$$З = З_{\text{осн}} + З_{\text{доп}} \quad (7)$$

где $З_{\text{осн}}$ – основная заработная плата;

$З_{\text{доп}}$ – дополнительная заработная плата (12-20 % от $З_{\text{осн}}$).

Основная заработная плата руководителя (лаборанта, инженера) от предприятия (при наличии руководителя от предприятия) рассчитывается по следующей формуле:

$$З_{\text{осн}} = З_{\text{дн}} \times T_p \quad (8)$$

где $З_{\text{осн}}$ – основная заработная плата одного работника;

T_p – продолжительность работ, выполняемых научно-техническим работником, раб. дн.;

$З_{\text{дн}}$ – среднедневная заработная плата работника, руб.

Среднедневная заработная плата рассчитывается по формуле:

$$З_{\text{дн}} = \frac{З_{\text{м}} \times M}{F_{\text{д}}} \quad (9)$$

где $З_{\text{м}}$ – месячный должностной оклад работника, руб.;

M – количество месяцев работы без отпуска в течение года: при отпуске в 24 рабочих дня $M = 11,2$ месяца для пятидневной рабочей недели;

$F_{\text{д}}$ – действительный годовой фонд рабочего времени, раб. дн.

Таблица 6 – Баланс рабочего времени

<i>Показатели рабочего времени</i>	<i>Начальник ОРТС</i>	<i>Программист1</i>	<i>Программист2</i>	<i>Программист3</i>	<i>Бакалавр</i>
Календарное число дней	365	365	365	365	365
Количество нерабочих дней - выходные дни - праздничные дни	118	118	118	118	118
Потери рабочего времени - отпуск - невыходы по болезни	24	24	24	24	24
Действительный годовой фонд рабочего времени	223	223	223	223	223

Месячный должностной оклад работника:

$$Z_m = Z_{тс} \times (1 + k_{пр} + k_d) \times k_p \quad (10)$$

где $Z_{тс}$ – заработная плата по тарифной ставке, руб.;

k_p – районный коэффициент, равный 1,3 (для города Томск);

$k_{пр}$ – премиальный коэффициент, равный 0,3 (т.е. 30% от $Z_{тс}$);

k_d – коэффициент доплат и надбавок составляет примерно 0,2 – 0,5.

Расчёт основной заработной платы сведен в таблицу 7.

Таблица 7 – Расчет основной заработной платы

Исполнитель	$Z_{тс}$	K_p	Z_m	$Z_{дн}$	T_p	$Z_{осн}$
Начальник ОРТС	25641,03	1,3	50000	1793.72	51.21	91856,40
Программист1	15384,62	1,3	30000	1076.35	135.57	145920,77
Программист2	15384,62	1.3	30000	1076.23	121.51	130772,71
Программист3	15384,62	1.3	30000	1076.23	138.23	148767,27
Бакалавр	15384,62	1,3	30000	1076.23	230.29	247845
<i>Итого:</i>						765162,15

4.2.2.4 Дополнительная заработная плата исполнителей

Расчет дополнительной заработной платы ведется по следующей формуле:

$$Z_{доп} = k_{доп} \times Z_{осн} \quad (11)$$

где $k_{доп}$ – коэффициент дополнительной заработной платы (на стадии проектирования принимается равным 0,12 – 0,15). Примем коэффициент равный 0,12.

Таблица 8 – Расчет дополнительной заработной платы

<i>Исполнитель</i>	<i>Основная заработная плата</i>	<i>$k_{доп}$</i>	<i>Дополнительная заработная плата</i>
Начальник ОРТС	91856,4	0,12	11022,77

<i>Исполнитель</i>	<i>Основная заработная плата</i>	<i>kдоп</i>	<i>Дополнительная заработная плата</i>
Программист1	145920,77	0,12	17510,49
Программист2	130772,71	0,12	15692,72
Программист3	148767,27	0,12	17852,07
Бакалавр	247845	0,12	29741,4
Итого	765162,15		91819,46

4.2.2.5 Отчисления во внебюджетные фонды (страховые отчисления)

В данной статье расходов отражаются обязательные отчисления по установленным законодательством Российской Федерации нормам органам государственного социального страхования (ФСС), пенсионного фонда (ПФ) и медицинского страхования (ФФОМС) от затрат на оплату труда работников.

Величина отчислений определяется по формуле:

$$Z_{\text{внеб}} = k_{\text{внеб}} \times (Z_{\text{осн}} + Z_{\text{доп}}) \quad (12)$$

где $k_{\text{внеб}}$ – коэффициент отчислений на уплату во внебюджетные фонды (пенсионный фонд, фонд обязательного медицинского страхования и пр.).

На 2018 г. в соответствии с положениями ст.58.2 закона №212-ФЗ установлены следующие тарифы страховых взносов: ПФР – 0.22 (22%), ФСС РФ – 0.029 (2,9%), ФФОМС – 0,051 (5,1%).

Таблица 9 – Отчисления во внебюджетные фонды

Исполнитель	Основная заработная плата	Коэффициент ПФРФ	Коэффициент ФСС	Коэффициент ФФОМС	Полная заработная плата
Начальник ОРТС	91856,4	22%	2,9	5,1	119413,32
Программист1	145920,77				189697
Программист2	130772,71				170004,52
Программист3	148767,27				193397,45
Бакалавр	247845				322198,5
Итого	765162,15				994710,79

4.2.2.6 Накладные расходы

Накладные расходы учитывают прочие затраты организации, не попавшие в предыдущие статьи расходов: печать и ксерокопирование материалов исследования, оплата услуг связи, электроэнергии, почтовые и телеграфные расходы, размножение материалов и т.д. Их величина определяется по следующей формуле:

$$Z_{\text{накл}} = (\text{сумма предыдущих статей}) \times k_{\text{нр}} \quad (13)$$

где $k_{\text{нр}}$ – коэффициент, учитывающий накладные расходы.

Величину коэффициента накладных расходов можно взять в размере 16%.

4.2.2.7 Формирование бюджета затрат научно-исследовательского проекта

Определение бюджета затрат на научно-исследовательский проект по каждому варианту исполнения приведен в таблице 10.

Таблица 10 – Расчет бюджета затрат

<i>Наименование статьи</i>	<i>Сумма, руб.</i>	<i>Примечание</i>
1. Материальные затраты	13423,75	Пункт 5.2.2.1
2. Затраты на специальное оборудование	65000	Пункт 5.2.2.2
3. Основная заработная плата исполнителей	765162,15	Пункт 5.2.2.3
4. Дополнительная заработная плата исполнителей	91819,46	Пункт 5.2.2.4
5. Отчисления во внебюджетные фонды	229548,64	Пункт 5.2.2.5

<i>Наименование статьи</i>	<i>Сумма, руб.</i>	<i>Примечание</i>
6. Затраты на научные и производственные командировки	0	-
7. Контрагентские расходы	0	-
8. Накладные расходы	186392,64	16 % от суммы ст. 1-7
9. Бюджет затрат	1351346,64	Сумма статей 1-8

Общий вывод по разделу

В результате работы по разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение» подсчитан бюджет затрат исполнения работ, который составляет 1351347 рублей.

Основными затратами являются трудовые затраты на исследование и разработку, поэтому для выполнения рассматриваемого проекта сэкономить на этом виде ресурса нельзя.

5 СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ

Разработка подсистем видеонаблюдения и мониторинга дорожной ситуации в автоматизированной системе управления дорожным движением производилась на предприятии АО ЭлеСи, в помещении офисного типа.

В данном разделе освещены вопросы производственной безопасности, экологической безопасности, безопасности в чрезвычайных ситуациях и правовые вопросы обеспечения безопасности, которые помогают свести к минимуму негативные последствия при разработке подсистем видеонаблюдения и мониторинга дорожной ситуации в автоматизированной системе управления дорожным движением. Также рассматривается вопрос пожарной безопасности и вопрос охраны окружающей среды.

5.1. Производственная безопасность

Все производственные факторы могут быть разделены на 4 группы [14]

- физические;
- химические;
- биологические;
- психофизиологические.

Биологические опасные и вредные производственные факторы, к которым можно отнести патогенные микроорганизмы и продукты их жизнедеятельности, а также микроорганизмы не оказывают на разработчика программного обеспечения особого влияния. Влияния также не оказывают химические опасные и вредные производственные факторы (токсические, раздражающие, сенсibiliзирующие, канцерогенные, влияющие на репродуктивную функцию, органы дыхания, желудочно-кишечный тракт и др.).

Влияние оказывают психофизиологические опасные и вредные факторы, а также физические опасные и вредные производственные факторы [Таблица 11].

Таблица 11 – Физические и психофизические опасные и вредные производственные факторы

Источник фактора, наименование видов работ	Факторы (по ГОСТ 12.0.003-74)		Нормативные документы
	Вредные	Опасные	
Работа с ПЭВМ, устройствами ввода-вывода информации	Повышенные или пониженные параметры микроклимата в помещениях		СанПиН 2.2.4.548-96 [] ГОСТ 30494-2011[]
	Недостаточная освещенность рабочей зоны. Отсутствие или недостаток естественного света		ГОСТ Р 55710-2013[] СанПиН 2.2.1/2.1.1.1278-03[]
	Повышенный уровень шума на рабочем месте		ГОСТ 12.1.003-2014[7] СН 2.2.4/2.1.8.562-96[8]
	Умственное перенапряжение		ТОИ Р-45-084-01[11]
		Опасность поражения электрическим током	ГОСТ Р 12.1.019-2009[9]

В стандарте также перечислен ряд других факторов, которые не будут рассмотрены в данной работе. Далее более подробно

5.1.1. Параметры микроклимата

В соответствии со стандартом [18] все помещения делятся на восемь различных категорий. Разработчик программного обеспечения выполняет свою работу в помещении, которое относится ко второй категории в соответствии с вышеприведенным стандартом, а именно помещение в котором люди заняты умственным трудом или учебой. Данный стандарт относит следующие параметры, которые характеризуют микроклимат в жилых и общественных помещениях:

- температура воздуха;
- скорость движения воздуха;
- относительная влажность воздуха;
- результирующая температура помещения;
- локальная асимметрия результирующей температуры.

Требования микроклимата для указанной категории помещения приведены в таблице 12.

Таблица 12 – Оптимальные и допустимые нормы температуры, относительной влажности и скорости движения воздуха в обслуживаемой зоне общественных и административных зданий

Период года	Категория помещения	Температура воздуха, °С		Результирующая температура, °С		Относительная влажность, %		Скорость движения воздуха, м/с	
		оптимальная	допустимая	оптимальная	допустимая	оптимальная	допустимая, не более	оптимальная, не более	допустимая, не более
Холодный	2	19-21	18-23	18-20	17-22	45-30	60	0,2	0,3
Теплый	2	23-25	18-28	22-24	19-27	60-30	65	0,15	0,25

Кроме параметров микроклимата стандарт также регламентирует качество воздуха, а также методы контроля, т.е. порядок проведения измерений показателей микроклимата. Стоит также отметить понятия холодный и теплый период года.

Теплый период года – период года, характеризующийся среднесуточной температурой наружного воздуха выше 8 °С.

Холодный период года – период года, характеризующийся среднесуточной температурой наружного воздуха, равной 8 °С и ниже.

Показатели микроклимата также определяются санитарными нормами и правилами. Разработчик программного обеспечения выполняет работы, которые относятся к категории Ia в соответствии с [28]. К данной категории относятся работы с интенсивностью энергозатрат до 120 ккал/ч (до 139 Вт), производимые сидя и сопровождающиеся незначительным физическим напряжением (ряд профессий на предприятиях точного приборо- и машиностроения, на часовом, швейном производствах, в сфере управления и т. п.).

Таблица 13 – Допустимые величины показателей микроклимата на рабочих местах производственных помещений

Период года	Категория работ по уровню энергозатрат	Температура воздуха, °С		Температура поверхностей, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с	
		Диапазон ниже оптимальных величин	Диапазон выше оптимальных величин			Для диапазона температур ниже оптимальных величин, не более	Для диапазона температур выше оптимальных величин, не более
Холодный	Ia(до 139)	20,0-21,9	24,1-25,0	19,0-26,0	15-75	0,1	0,1
Теплый	Ia(до 139)	21,0-22,9	25,1-28,0	20,0-29,0	15-75	0,1	0,2

В офисном помещении, в котором производится разработка программного обеспечения температура воздуха в холодный период года составляет 20,9 °С при относительной влажности воздуха 32%. В теплый период измерения не проводились. Также не проводились измерения скорости движения воздуха в связи с невозможностью проведения данных измерений.

В данной части рассмотрены основные нормы и правила, регулирующие параметры микроклимата. Офисное помещение АО ЭлеСи, где производится разработка, соответствует требованиям санитарных правил и норм, а также стандарту ГОСТ 30494-2011.

5.1.2. Освещенность рабочей зоны

В соответствии с [16] помещение для разработки можно охарактеризовать как административное здание с наименованием зрительной работы и видом деятельности – письмо, машинопись, чтение, обработка данных. Для данной категории помещений устанавливаются следующие нормы:

эксплуатационная освещенность ($E_{\text{экс}}$, лк) – 500 лк;

зона освещенности непосредственного окружения ($E_{\text{ср}}$, лк) – 300 лк;

равномерность освещенности (U_0) – не менее 0.6;

коэффициент пульсации освещенности (K_p , %) – не более 10%.

Коэффициенты отражения окружающих поверхностей должны быть:

- от 0,7 до 0,9 - для потолков;
- от 0,5 до 0,8 - для стен;
- от 0,2 до 0,7 - для рабочих поверхностей;
- от 0,2 до 0,4 - для пола.

Нормы СанПиН также определяют требования к естественному, искусственному и совмещенному освещению. Согласно [25] офисное помещение АО ЭлеСи относится к помещению для работы с дисплеями и видеотерминалами, залам ЭВМ. Для таких помещений предъявляются следующие требования:

- рабочая поверхность – горизонтальная;
- высота рабочей поверхности – 0,8 метра;
- расположение экрана монитора – вертикально относительно пола;
- высота расположения монитора – 1,2 метра;
- освещенность – 400 лк;
- коэффициент пульсации – не более 10%.

В помещении, где производится разработка освещение является общим, потолок подвесной, выполнен в виде потолочной плитки из негорючих материалов, стены выкрашены водно-дисперсионной краской белого цвета. Высота рабочей поверхности (стола) составляет 0,8 метра над уровнем пола, цвет стола коричневый. Пол покрыт паркетом коричневого цвета. Все вышеперечисленные цвета имеют коэффициенты отражений в заданных стандартом диапазонах.

По периметру равномерно расположены 15 светильников ЛПО. В каждом светильнике расположено по 4 источника света типа ЛБ-40. Габариты помещения составляют: длина – 10 метров; ширина – 15,5 метров; высота потолка – 3 метра.

Коэффициент пульсации можно оценить в соответствии с [22]. Так в соответствии с методическими указаниями, типом используемых светильников, предельными расстояниями между светильниками, условиях при которых соблюдаются нормированные условия можно прийти к выводу, что

коэффициент пульсации составляет 15%, что немного превышает требования стандарта, однако в требованиях также есть пометка о том, что коэффициент пульсации не нормируется в помещениях при отсутствии условий для возникновения стробоскопического эффекта. Условия возникновения данного эффекта отсутствуют в исследуемом помещении в связи с отсутствием вращающихся, двигающихся элементов, частота которых может совпадать с частотой пульсации ламп.

Необходимый световой поток лампы рассчитывается:

$$\Phi = \frac{E_n \times S \times k \times z}{N \times \eta \times n} \quad (14)$$

где Φ – световой поток от лампы;

E_n – норма освещенности;

k – коэффициент запаса (1,2 для газоразрядных ламп);

z – коэффициент неравномерности освещения (1,1 для люминесцентных ламп);

N – количество светильников;

η – коэффициент использования светового потока;

n – число ламп в светильнике.

Для нахождения коэффициента использования светового потока необходимо определить индекс помещения:

$$i = \frac{S}{(a + b) \times h} \quad (15)$$

где S – площадь помещения;

a – длина помещения;

b – ширина помещения;

h – высота над рабочей областью помещения.

После определения индекса помещения, в соответствии с типом светильников, и коэффициентов отражения потолка, стен и пола, по таблице определяется коэффициент использования светового потока для конкретного

типа ламп. Коэффициент помещения $i = 2.76$, тогда коэффициент использования для ЛБ-40 составляет 0,6.

Подставив все исходные данные и найдя необходимое количество светильников при световом потоке каждой лампы $\Phi=2800$ лм, минимально необходимое количество светильников данного типа составляет 14. В связи с этим можно сделать вывод, что освещенность рабочей зоны соответствует нормам.

5.1.3. Уровень шума

Шум в соответствии с [15] - звуковые колебания в диапазоне слышимых частот, способные оказать вредное воздействие на безопасность и здоровье работника.

На работодателе лежит основная ответственность за обеспечение безопасности при воздействии шума на работников. В первую очередь, он должен обеспечить посредством принятия соответствующих мер соблюдение гигиенических нормативов и снижение риска, связанного с воздействием шума на работников. Эти меры могут включать в себя, в частности:

- оценку риска потери слуха работником;
- проектирование рабочих мест с учетом допустимого уровня риска;
- использование малошумных машин;
- контроль правильности использования средств индивидуальной защиты от шума;
- организацию профилактических мероприятий, ослабляющих неблагоприятное воздействие шума и ряд других.

Работник, оповещенный работодателем о возможных рисках, связанных с воздействием шума, и о необходимости использовать в целях снижения рисков средства индивидуальной защиты от шума, должен следовать установленным работодателем правилам безопасного ведения работ и применения средств индивидуальной защиты от шума. Ответственность за последствия отказа

следовать установленным правилам и применять предписанные средства индивидуальной защиты лежит на работнике.

Шум на рабочих местах кроме стандарта также регулируется санитарными нормами [29], согласно которым устанавливаются следующие предельно допустимые уровни звука на рабочих местах [Таблица 14].

Таблица 14 – Предельно допустимые уровни звука на рабочих местах для разных категорий тяжести и напряженности в дБА

<i>Категория напряженности трудового процесса</i>	<i>Категория тяжести трудового процесса</i>				
	<i>Легкая физическая нагрузка</i>	<i>Средняя физическая нагрузка</i>	<i>Тяжелый труд первой степени</i>	<i>Тяжелый труд второй степени</i>	<i>Тяжелый труд третьей степени</i>
Напряженность легкой степени	80	80	75	75	75
Напряженность средней степени	70	70	65	65	65
Напряженный труд первой степени	60	60			
Напряженный труд второй степени	50	50			

Однако при работе с ПЭВМ действуют требования [26] и уровень шума не должен превышать 50 дБА [Таблица 16].

5.1.4 Умственное перенапряжение

Для видов трудовой деятельности устанавливается 3 категории тяжести и напряженности работы с компьютером, которые определяются [Таблица 15]:

- для группы А - по суммарному числу считываемых знаков за рабочую смену (не более 60000 знаков за смену);

- для группы Б - по суммарному числу считываемых или вводимых знаков за рабочую смену (не более 40000 знаков за смену);

- для группы В - по суммарному времени непосредственной работы с компьютером за рабочую смену (не более 6 часов за смену).

Таблица 15 – Категория работы по тяжести и напряженности

<i>Категория работы по тяжести и напряженности</i>	<i>Уровень нагрузки за рабочую смену при видах работы на ПК</i>		
	<i>Группа А Количество знаков</i>	<i>Группа Б Количество знаков</i>	<i>Группа В Время работы, ч</i>
III	До 60000	До 40000	До 6,0

При 8-часовой рабочей смене и работе на ПК регламентированные перерывы для категории работ III следует устанавливать в соответствии с [30] через 1,5- 2,0 часа от начала рабочей смены и через 1,5-2,0 часа после обеденного перерыва продолжительностью 20 минут каждый или продолжительностью 15 минут через каждый час работы. На предприятии, где производится разработка программного обеспечения перерывы установлены каждые 2 часа, продолжительностью 20 минут каждый.

5.1.5. Электробезопасность

Опасное и вредное воздействия на людей электрического тока, электрической дуги и электромагнитных полей проявляются в виде электротравм и профессиональных заболеваний.

Степень опасного и вредного воздействия на человека электрического тока, электрической дуги и электромагнитных полей согласно [18], зависит от:

- рода и величины напряжения и тока;
- частоты электрического тока;
- пути тока через тело человека;
- продолжительности воздействия электрического тока или электромагнитного поля на организм человека;
- условий внешней среды.

Электробезопасность должна обеспечиваться:

- конструкцией электроустановок;
- техническими способами и средствами защиты;

- организационными и техническими мероприятиями.

Электроустановки и их части должны быть выполнены таким образом, чтобы работающие не подвергались опасным и вредным воздействиям электрического тока и электромагнитных полей, и соответствовать требованиям электробезопасности.

Требования безопасности при эксплуатации электроустановок на производстве должны соответствовать нормативным требованиям охраны труда, содержащимся в Федеральных законах и иных нормативных правовых актах Российской Федерации [27].

Требования безопасности при пользовании электроустановками бытового назначения должны содержаться в прилагаемых к ним инструкциях по эксплуатации предприятий-изготовителей

Для обеспечения защиты от случайного прикосновения к токоведущим частям необходимо применять следующие способы и средства:

- защитные оболочки;
- защитные ограждения (временные или стационарные);
- изоляцию токоведущих частей
- изоляцию рабочего места;
- защитное отключение;
- предупредительную сигнализацию, знаки безопасности и др.

Для обеспечения защиты от поражения электрическим током при прикосновении к металлическим нетоковедущим частям, которые могут оказаться под напряжением в результате повреждения изоляции, применяют следующие способы:

- защитное заземление;
- зануление;
- систему защитных проводов;
- защитное отключение;
- изоляцию нетоковедущих частей;
- электрическое разделение сети;

средства индивидуальной защиты и др.

Офисное помещение, в котором проводятся работы по категории опасности поражения электрическим током относится к помещениям без повышенной опасности, однако при работе с ПЭВМ необходимо соблюдение инструкции по охране труда при работе на персональном компьютере [30].

5.2. Экологическая безопасность

В последние десятилетия выросло производство продукции электротехнической и электронной промышленности. В результате вывода из эксплуатации эта продукция превращается в отходы, которые содержат токсичные вещества, представляющие собой существенную угрозу для окружающей среды, жизни и здоровья людей.

Отработавшее электротехническое и электронное оборудование (ОЭЭО) подразделяется на две основные категории в соответствии с [20]:

- оборудование, которое может быть повторно использовано.
- оборудование, которое может быть переработано для целей получения вторичных материальных и энергетических ресурсов.

Для оценки возможности повторного использования электротехнического и электронного оборудования необходимо провести проверку функционального состояния, соответствующую проверке, которая осуществляется перед распространением нового электротехнического или электронного оборудования соответствующего типа среди конечных пользователей.

Сбор, хранение, транспортирование и разборка оборудования осуществляется в соответствии с общими требованиями безопасности при обращении с отходами соответствующих составов и классов опасности с учетом положений приложения Б [20].

Сбор, хранение, транспортирование и разборку оборудования могут осуществлять следующие хозяйствующие субъекты:

- производители электротехнического и электронного оборудования;
- предприятия по переработке ОЭЭО;
- специализированные пункты сбора и хранения ОЭЭО;
- пункты сбора вторичного сырья.

При сборе, хранении и транспортировании ОЭЭО необходимо обеспечить условия, позволяющие сохранить неизменность свойств ОЭЭО или обеспечение их изменения в пределах, допускаемых производителем для соответствующего этапа жизненного цикла ОЭЭО.

Сбор и хранение ОЭЭО организуют так, чтобы перемещение ОЭЭО могло быть осуществлено с помощью подручных технических средств (тележек, конвейеров, рабочих столов и т.д.).

Таким образом, для уменьшения загрязнений окружающей среды, а также оказания негативного воздействия на жизнь и здоровье людей необходимо соблюдать все этапы по безопасному сбору, хранению, транспортированию и разборке отработавшего электротехнического и электронного оборудования в соответствии с [20]. Либо все этапы технологического цикла отходов в соответствии с [19].

5.3. Безопасность в чрезвычайных ситуациях

Все работники предприятия должны допускаться к работе после прохождения инструктажа и обучения мерам пожарной безопасности. Обучение работников мерам пожарной безопасности осуществляется путем проведения противопожарного инструктажа и прохождения пожарно-технического минимума в соответствии с нормативными документами по пожарной безопасности.

В целях реализации Приказа МЧС РФ от 12.12.2007 N 645 [24] инструктажи по пожарной безопасности подразделяются на:

- вводный;
- повторный (не реже 1 раза в год);

- внеплановый;
- целевой.

Работник организации в соответствии с инструкцией о мерах пожарной безопасности в зданиях и помещениях организации [21] обязан:

- соблюдать требования пожарной безопасности, установленные в организации;
- знать и уметь пользоваться первичными средствами пожаротушения;
- выполнять требования пожарной безопасности, применимо к своему рабочему месту, обеспечить ежедневную уборку материалов, оборудования и приспособлений;
- при обнаружении нарушений в работе немедленно уведомлять об этом своего непосредственного руководителя;
- знать контактные номера телефонов для вызова пожарной охраны, до прибытия пожарной охраны принимать посильные меры по спасению людей, имущества;
- оказывать содействие пожарной охране при тушении пожаров;
- уметь пользоваться первичными средствами пожаротушения;
- своевременно проходить инструктажи по пожарной безопасности, а также обучение по пожарно-техническому минимуму;
- выполнять предписания, постановления и иные законные требования специалиста по охране труда и руководителей организации.

Все здания и сооружения должны удовлетворять нормам и правилам пожарной безопасности зданий и сооружений [32]. Однако также стоит отметить что исключение из применения СНиП составляют постройки специального назначения (склады взрывчатых веществ, объекты военного назначения, подземные сооружения метрополитенов).

В общественных зданиях и сооружениях на каждом этаже должно размещаться не менее двух переносных огнетушителей. Огнетушители следует располагать на видных местах вблизи от выходов из помещений на высоте не

более 1,35 м. Размещение первичных средств пожаротушения в коридорах, переходах не должно препятствовать безопасной эвакуации людей [23].

В данном разделе рассмотрены основные вопросы пожарной безопасности, в том числе обязанности работника в соответствии с инструкцией о мерах пожарной безопасности в зданиях и помещениях организации, вопрос санитарных норм и правил, которым должны удовлетворять здания и сооружения, а также основные поражающие факторы.

5.4 Правовые и организационные вопросы обеспечения безопасности

При проектировании системы был заключен договор подряда с АО ЭлеСи в соответствии с [31]. Основные отличия гражданско-правового договора подряда от трудового следующие:

- по гражданско-правовому договору подряда подрядчик обязан выполнить конкретное задание заказчика, которое известно уже на момент заключения договора;
- сдача-приемка выполненных работ оформляется актом;
- подрядчик сам определяет порядок выполнения возложенных на него договором обязанностей и исполняет их за свой счет;
- подрядчик вправе привлечь к исполнению своих обязанностей третьих лиц;
- подрядчик получает не заработную плату, а предусмотренное договором вознаграждение.

Кроме трудового кодекса работодатель обязан соблюдать гигиенические требования и нормы при организации рабочего места.

ПЭВМ должны соответствовать требованиям в соответствии с [26], и каждый их тип подлежит санитарно-эпидемиологической экспертизе с оценкой в испытательных лабораториях, аккредитованных в установленном порядке. Допустимые уровни звукового давления и уровней звука, не должны превышать значений [Таблица 16].

Таблица 16 – Допустимые значения уровней звукового давления в октавных полосах частот и уровня звука, создаваемого ПЭВМ

Уровни звукового давления в октавных полосах со среднегеометрическими частотами									Уровни звука в дБА
31,5 Гц	63 Гц	125 Гц	250 Гц	500 Гц	1000 Гц	2000 Гц	4000 Гц	8000 Гц	
86 дБ	71 дБ	61 дБ	54 дБ	49 дБ	45 дБ	42 дБ	40 дБ	38 дБ	50

Временные допустимые уровни электромагнитных полей (ЭМП), создаваемых ПЭВМ, не должны превышать значений [Таблица 17].

Таблица 17 – Временные допустимые уровни ЭМП

Наименование параметров		ВДУ ЭМП
Напряженность электрического поля	В диапазоне частот 5 Гц – 2 кГц	25 В/м
	В диапазоне частот 2 кГц – 400 кГц	2,5 В/м
Плотность магнитного потока	В диапазоне частот 5 Гц – 2 кГц	250 нТл
	В диапазоне частот 2 кГц – 400 кГц	25 нТл
Электростатический потенциал видеомонитора	экрана	500 В

Кроме перечисленных выше параметров, которые регулирует [31] также данные правила устанавливают ряд других важных норм, в том числе требования к помещениям, к микроклимату, общим требованиям к организации рабочих мест, а также суммарное время регламентированных перерывов в зависимости от продолжительности работы, вида и категории трудовой деятельности с ПЭВМ.

Общий вывод по разделу

При выполнении раздела «Социальная ответственность» работник обеспечен средствами защиты от вредных факторов и соблюдены все правовые и организационные нормы.

Заключение

В первую очередь стоит отметить, что представленные результаты разработанных подсистем являются не единственными составляющими программного комплекса «АвтоМагистраль». Кроме рассмотренных подсистем в состав интеллектуальной транспортной системы входят и другие:

- AM.Meteo;
- AM.Monitoring;
- AM.WebHMI;
- AM.Sender
- AM.Emer;
- AM.Messages;
- AM.Aid;
- AM.GeoSystem.

Реализация проекта направлена на удовлетворение интересов потенциально заинтересованного заказчика – "ГК Автодор", представители которого на встрече в компании ЭлеСи в Томске и во время презентаций программного комплекса на профильных мероприятиях (IV МЕЖДУНАРОДНАЯ КОНФЕРЕНЦИЯ «Роль и место интеллектуальных транспортных систем в сети платных автомобильных дорог Российской Федерации. Современные тенденции развития» г. Санкт-Петербург) отметили ряд перспективных решений, которые могут быть востребованы и применены на дорогах, по мере реализации ими инновационных проектов.

Достигнут ряд предварительных договоренностей с Государственной компанией «Российские автомобильные дороги» о рассмотрении интеграционной платформы в качестве возможного варианта решения на вновь проектируемых и реконструируемых участках автомагистралей.

Список использованных источников

1. Взаимодействие управляемого и не управляемого кода. Российский журнал для программистов RSDN [Электронный ресурс]. – 2009. – Режим доступа: http://rstdn.org/article/dotnet/Native_Managed_Interop.xml
2. Вигерс К.И. Разработка требований к программному обеспечению / К.И. Вигерс. – М.: Русская редакция Microsoft, 2004. – 575 с.
3. Деннинг А. ActiveX для профессионалов / А. Деннинг. – Спб.: Питер, 1997. – 624 с.
4. Евстигнеев И.А. Интеллектуальные транспортные системы на автомобильных дорогах федерального значения России. / И.А. Евстигнеев. – М.: Издательство Перо, 2015. – 164 с.
5. Евстигнеев, И.А. Основы создания интеллектуальных транспортных систем на автомобильных дорогах федерального значения России. / И.А. Евстигнеев. – М.: Издательство Перо, 2016. – 260 с.
6. Ксенчук Е. Системное мышление. Границы ментальных моделей и системное видение мира / Е. Ксенчук. – М.: Дело, 2011. – 370 с.
7. Купер А. Об интерфейсе. Основы проектирования взаимодействия / А. Купер, Р. Рейман, Д. Кронин. – Спб.: «Символ», 2009. – 687с.
8. Мартин Р. Принципы, паттерны и методики гибкой разработки на языке С# / Р. Мартин, М. Мартин. – Спб.: Символ-Плюс, 2011. – 768с.
9. Сборки .NET. Руководство по С#. Программирование на языке С# [Электронный ресурс]. – 2017. – Режим доступа: https://professorweb.ru/my/csharp/assembly/level1/assembly_index.php
10. Calling Managed .NET C# COM Objects from Unmanaged C++ Code. Code project / [Электронный ресурс]. – 2006. – Режим доступа: <https://www.codeproject.com/Articles/12673/Calling-Managed-NET-C-COM-Objects-from-Unmanaged-C>

11. C# ActiveX control: Центр разработки для Windows [Электронный ресурс]. – 2012. – Режим доступа: <https://code.msdn.microsoft.com/windowsapps/CSActiveX-c3f43e7d>
12. Griffiths I. Programming C# 5.0. Building Windows 8, Web and Desktop Applications for the .NET 4.5 Framework / I. Griffiths. – O'Reilly, 2012. – 886 p
13. Nathan A. .NET and COM: The complete interoperability guide / A. Nathan. – USA, Indiana.: SAMS, 2009 – 1608 p
14. ГОСТ 12.0.003-74. Система стандартов безопасности труда. Опасные и вредные производственные факторы. Классификация. – М.: Изд-во стандартов, 1974. – 4 с
15. ГОСТ 12.1.003-2014. Система стандартов безопасности труда. Шум. Общие требования безопасности. – М.: Стандартинформ, 2015. – 24 с
16. ГОСТ 24940-2016. Здания и сооружения. Методы измерения освещенности. – М.: Стандартинформ, 2016. – 24
17. ГОСТ 30494-2011. Здания жилые и общественные. Параметры микроклимата в помещениях. – М.: Стандартинформ, 2013. – 15 с
18. ГОСТ Р 12.1.019-2009. Система стандартов безопасности труда. Электробезопасность. Общие требования и номенклатура защиты. – М.: Стандартинформ, 2010. – 32 с
19. ГОСТ Р 53692-2009. Ресурсосбережение. Обращение с отходами. Этапы технологического цикла отходов. – М.: Стандартинформ, 2011. – 20 с
20. ГОСТ Р 55102-2012. Ресурсосбережение. Обращение с отходами. Руководство по безопасному сбору, хранению, транспортированию и разработке отработавшего электротехнического и электронного оборудования, за исключением ртутьсодержащих устройств и приборов. – М.: Стандартинформ, 2014. – 16 с
21. Инструкция о мерах пожарной безопасности на территориях, зданиях и помещениях организации № 01-15
22. Методические указания МУ 2.2.4.706-98ОМ/МУ ОТ РМ 01-98. Оценка освещения рабочих мест

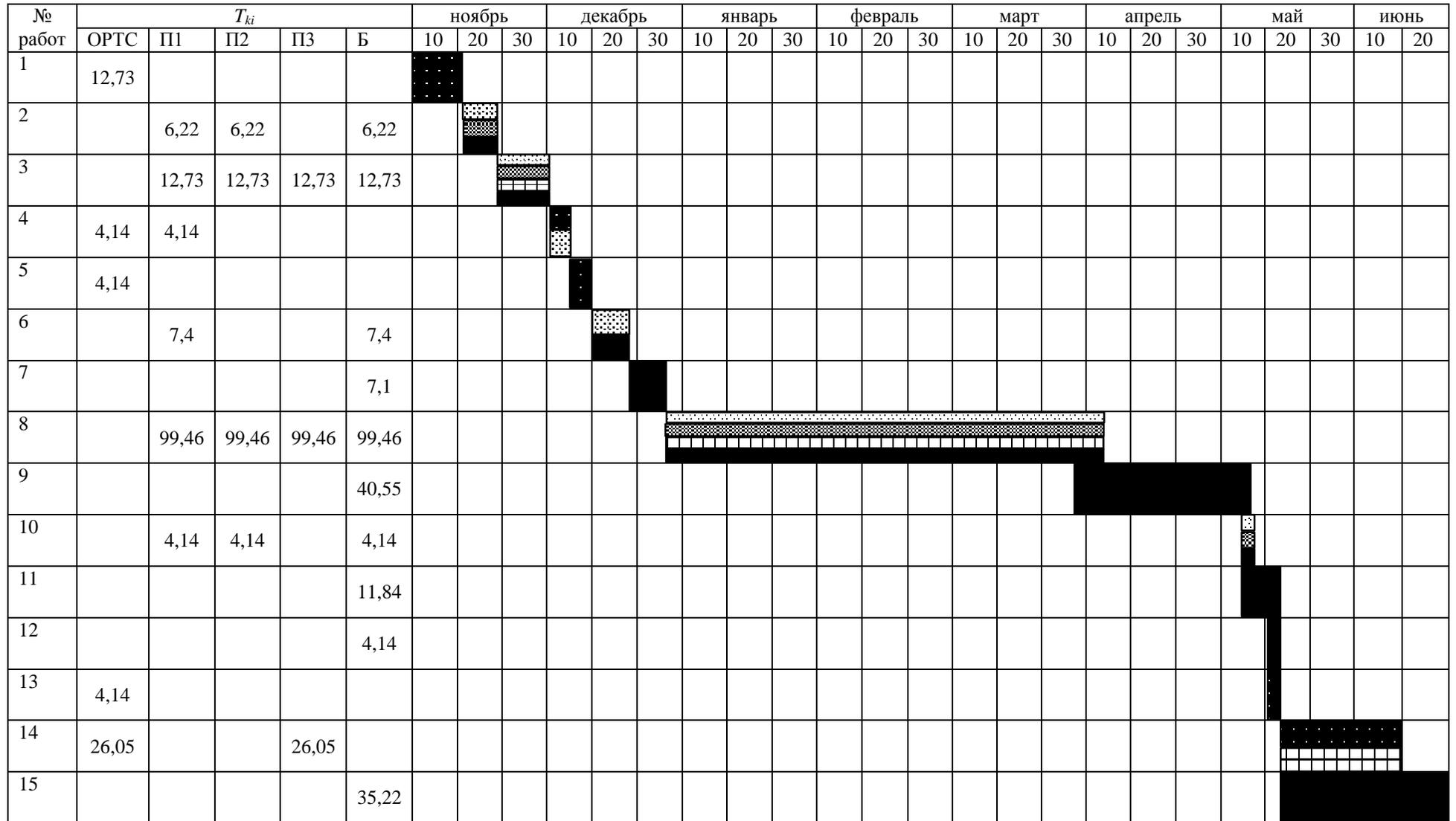
23. ППБ 139-87. Правила пожарной безопасности для энергетических предприятий
24. Приказ МЧС РФ от 12.12.2007 N 645 – Об утверждении норм пожарной безопасности. Обучение мерам пожарной безопасности работников организаций
25. СанПиН 2.2.1/2.1.1.1278-03. Гигиенические требования к естественному, искусственному и совмещенному освещению жилых и общественных зданий
26. СанПиН 2.2.2/2.4.1340-03. Гигиенические требования к персональным электронно-вычислительным машинам и организации работы
27. СанПиН 2.2.4.1191-03. Электромагнитные поля в производственных условиях
28. СанПиН 2.2.4.548-96. Физические факторы производственной среды. Гигиенические требования к микроклимату производственных помещений. – М.: Минздрав России, 1997
29. СН 2.2.4/2.1.8.562-96. Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории жилой застройки
30. ТОИ Р-45-084-01. Типовая инструкция по охране труда при работе на персональном компьютере
31. Трудовой кодекс Российской Федерации № 197-ФЗ от 30.12.2001
32. Строительные нормы и правила СНиП 21-01-97. Пожарная безопасность зданий и сооружений. – М.: Центр проектной продукции в строительстве, 2007. – 21 с

Приложение А

Таблица А1 – Временные показатели проведения различных этапов работ

№ работ	Трудоемкость работ			Исполнители	Длительность работ в рабочих днях					Длительность работ в календарных днях				
	t_{min} , чел.-дни	t_{max} , чел.-дни	$t_{ож}$, чел.-дни		ОРТС	Прогр.1	Прогр.2	Прогр.3	Бакалавр	ОРТС	Прогр.1	Прогр.2	Прогр.3	Бакалавр
1	7	11	8,6	ОРТС	8,6					12,73				
2	12	21	15,6	П1, П2, Б		5,2	5,2		5,2		6,22	6,22		6,22
3	20	56	34,4	П1, П2, П3, Б		8,6	8,6	8,6	8,6		12,73	12,73	12,73	12,73
4	4	8	5,6	ОРТС, П1	2,8	2,8				4,14	4,14			
5	2	4	2,8	ОРТС	2,8					4,14				
6	6	16	10	П1, Б		5			5		7,4			7,4
7	4	6	4,8	Б					4,8					7,1
8	240	312	268,8	П1, П2, П3, Б		67,2	67,2	67,2	67,2		99,46	99,46	99,46	99,46
9	21	37	27,4	Б					27,4					40,55
10	6	12	8,4	П1, П2, Б		2,8	2,8		2,8		4,14	4,14		4,14
11	6	11	8	Б					8					11,84
12	2	4	2,8	Б					2,8					4,14
13	2	4	2,8	ОРТС	2,8					4,14				
14	28	46	35,2	ОРТС, П3	17,6			17,6		26,05			26,05	
15	21	28	23,8	Б					23,8					35,22
Всего					34,6	91,6	83,8	93,4	155,6	51,21	135,57	121,51	138,23	230,29

Таблица А2 – Линейный график работ



■ - Начальник ОРТС; ■ - Программист1; ■ - Программист2; ■ - Программист3; ■ - Бакалавр

Примеры работы подсистемы АМ.Еye

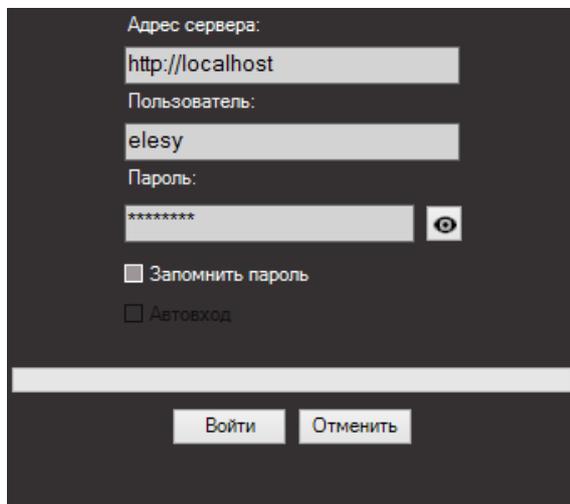


Рисунок Б1 - Авторизация АМ.Еye – Vig

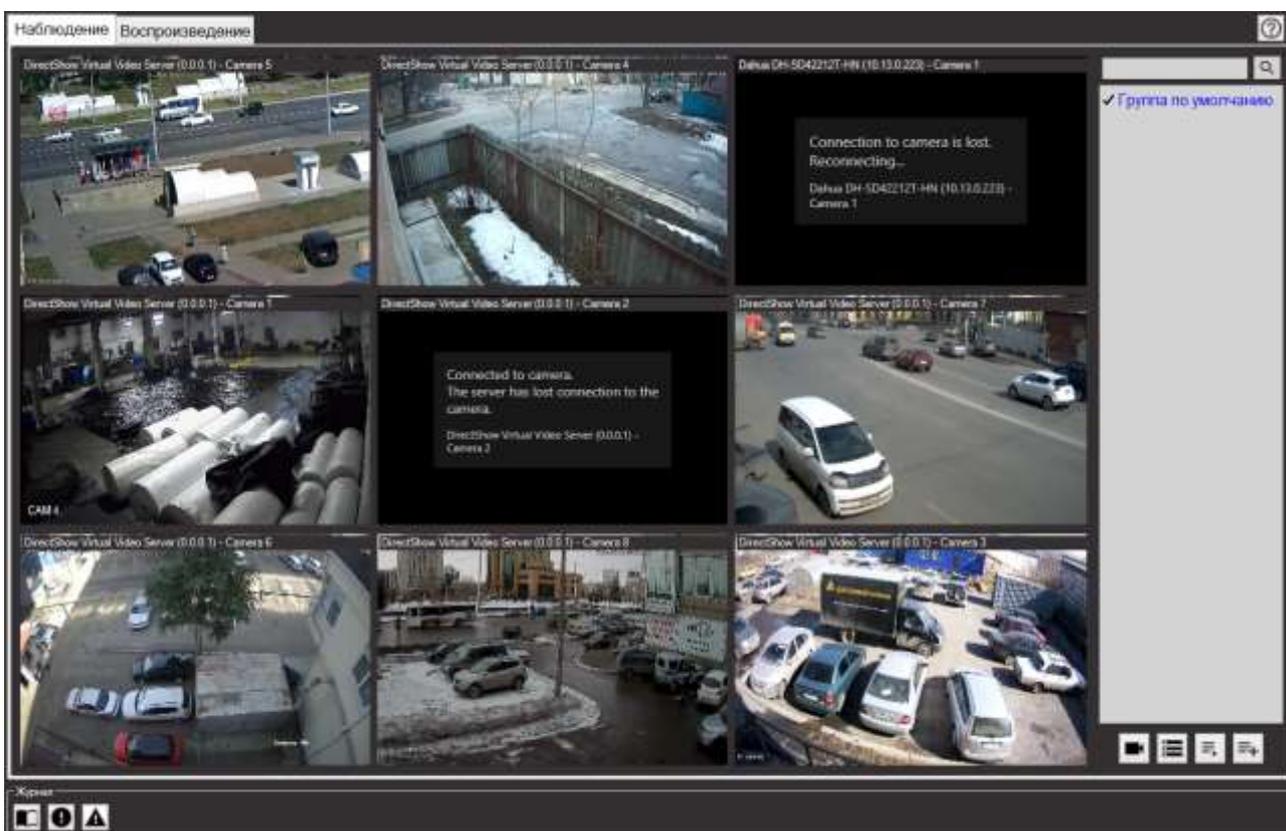


Рисунок Б2 – Режим наблюдения компонента АМ.Еye – Vig

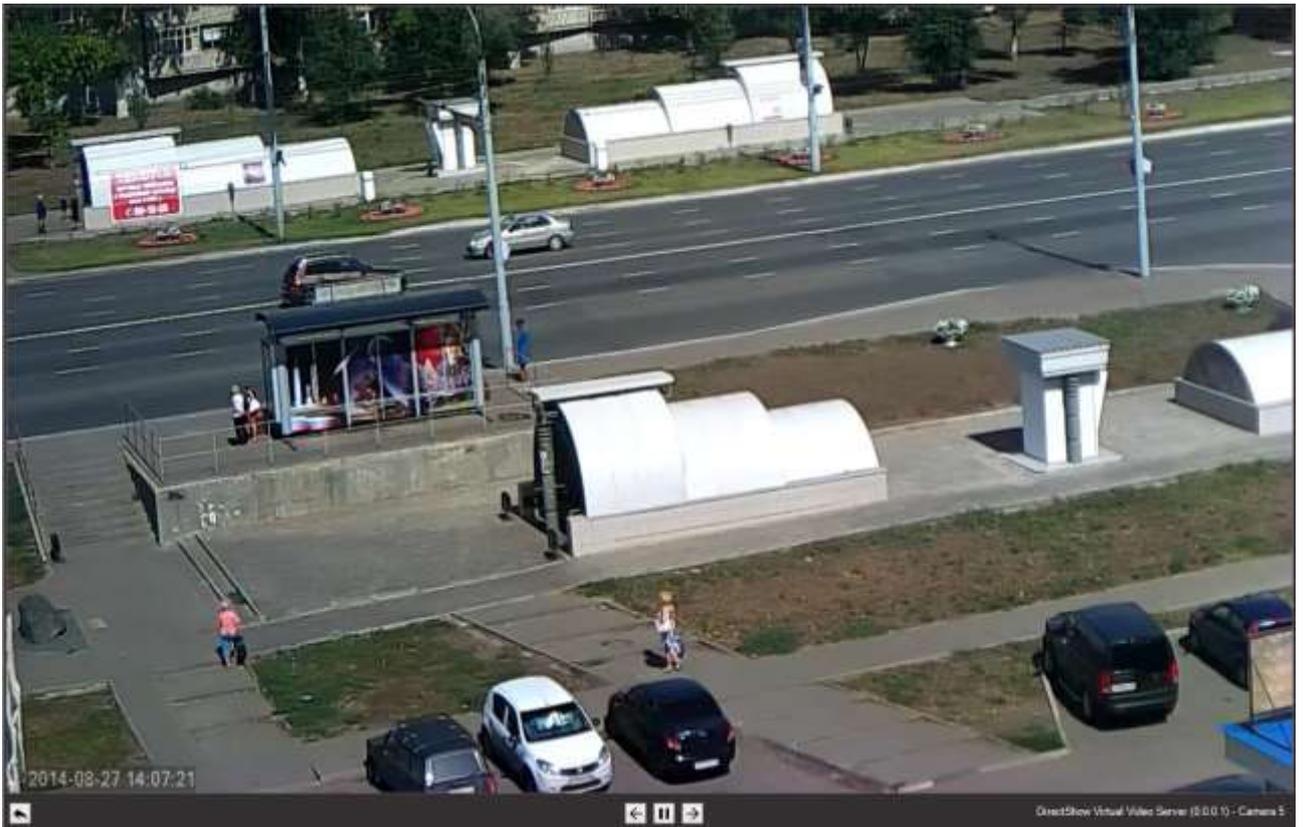


Рисунок Б3 – Режим воспроизведения по сценарию

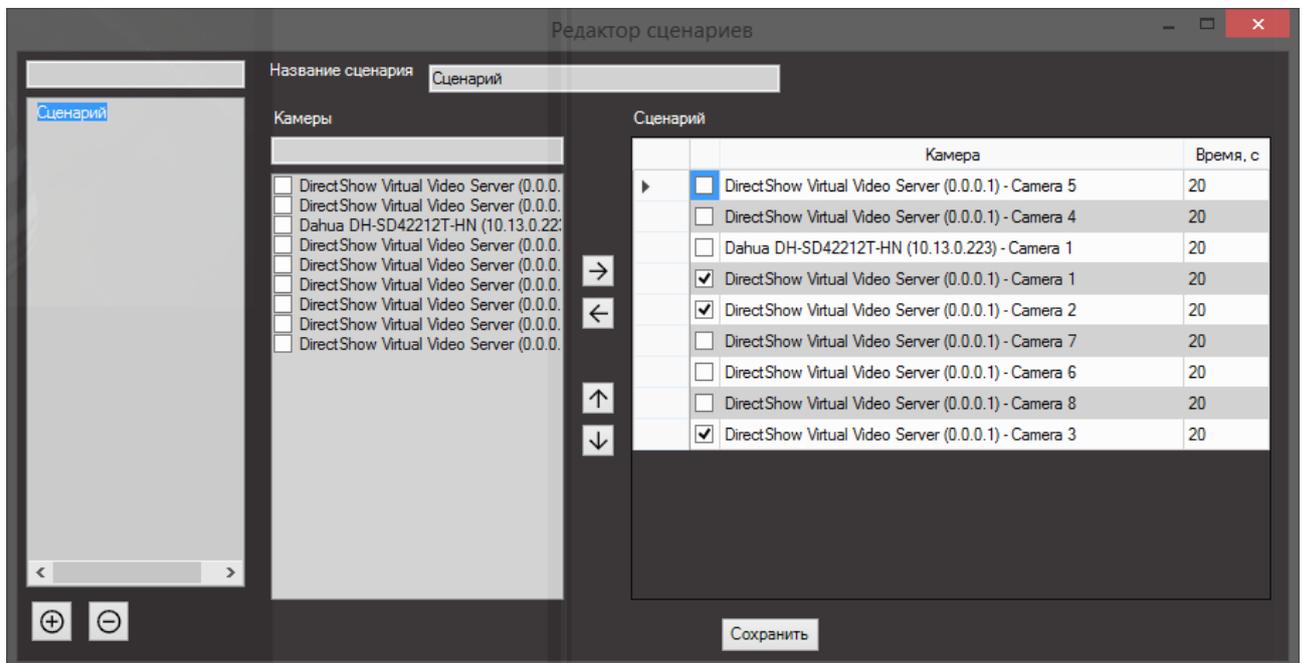


Рисунок Б4 – Редактор сценариев компонента AM.Eye – Big

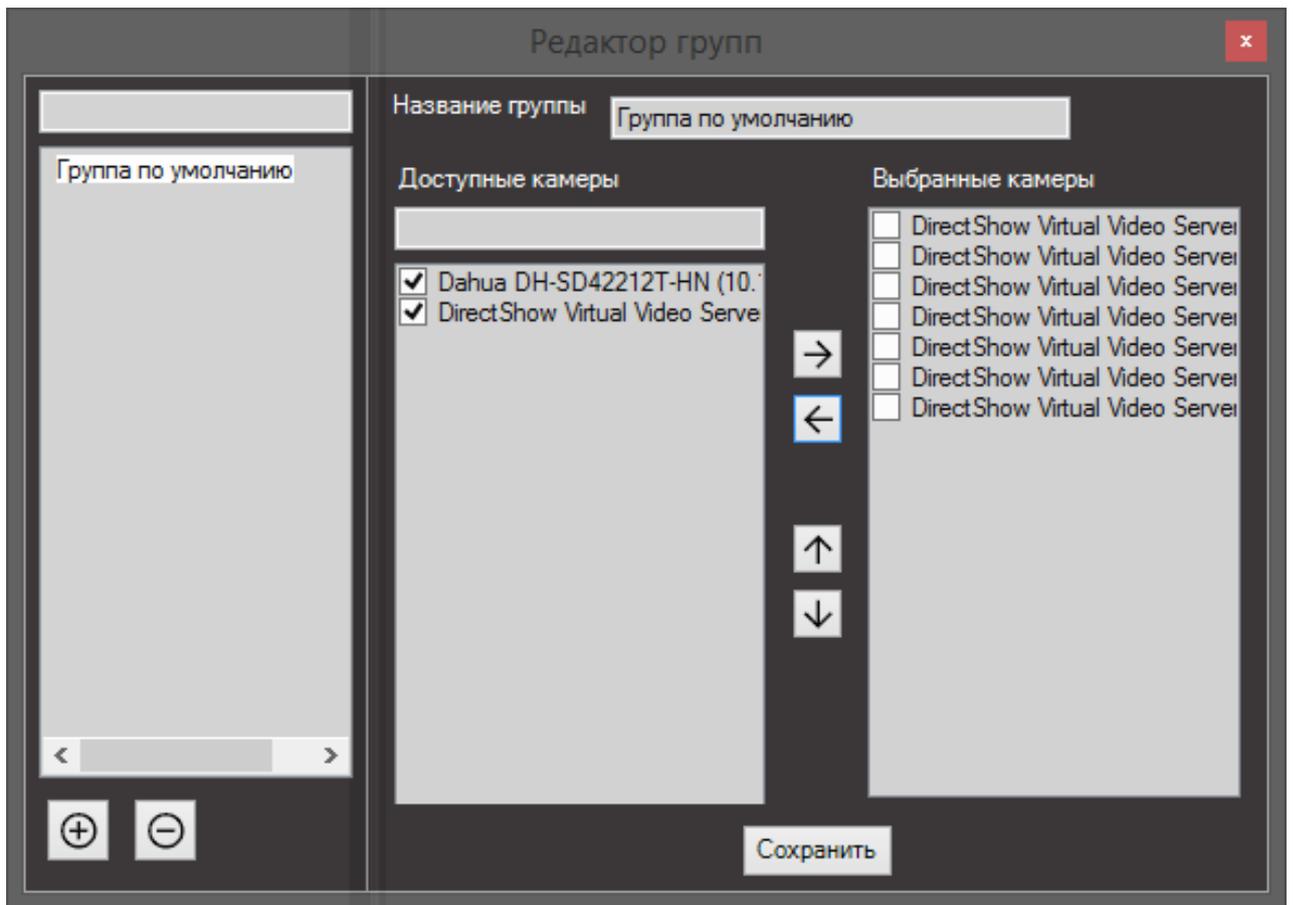


Рисунок Б5 – Редактор групп компонента AM.Eye – Big

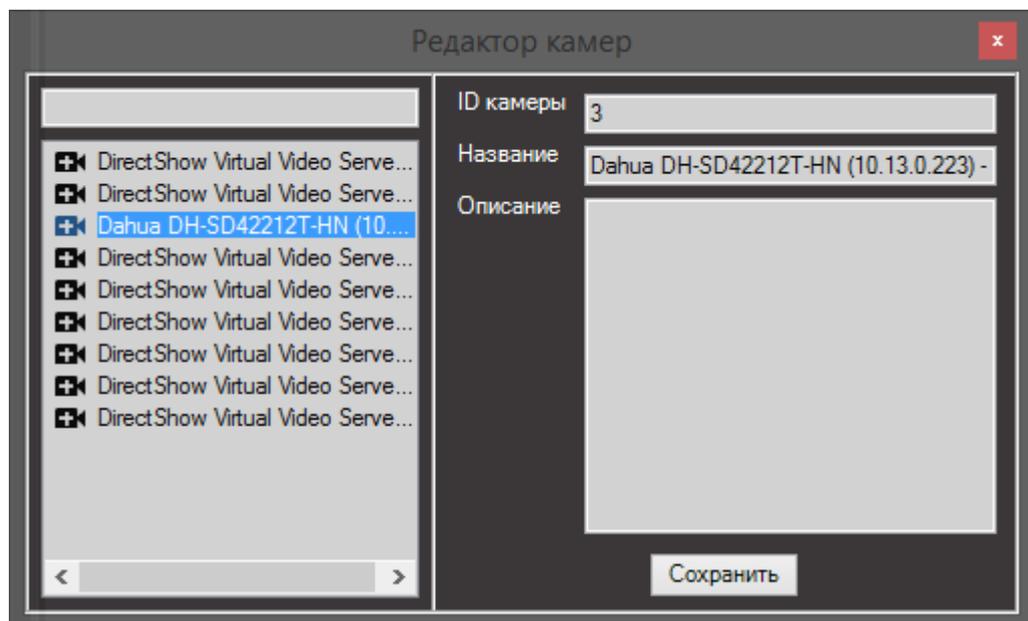


Рисунок Б6 – Редактор камер компонента AM.Eye – Big

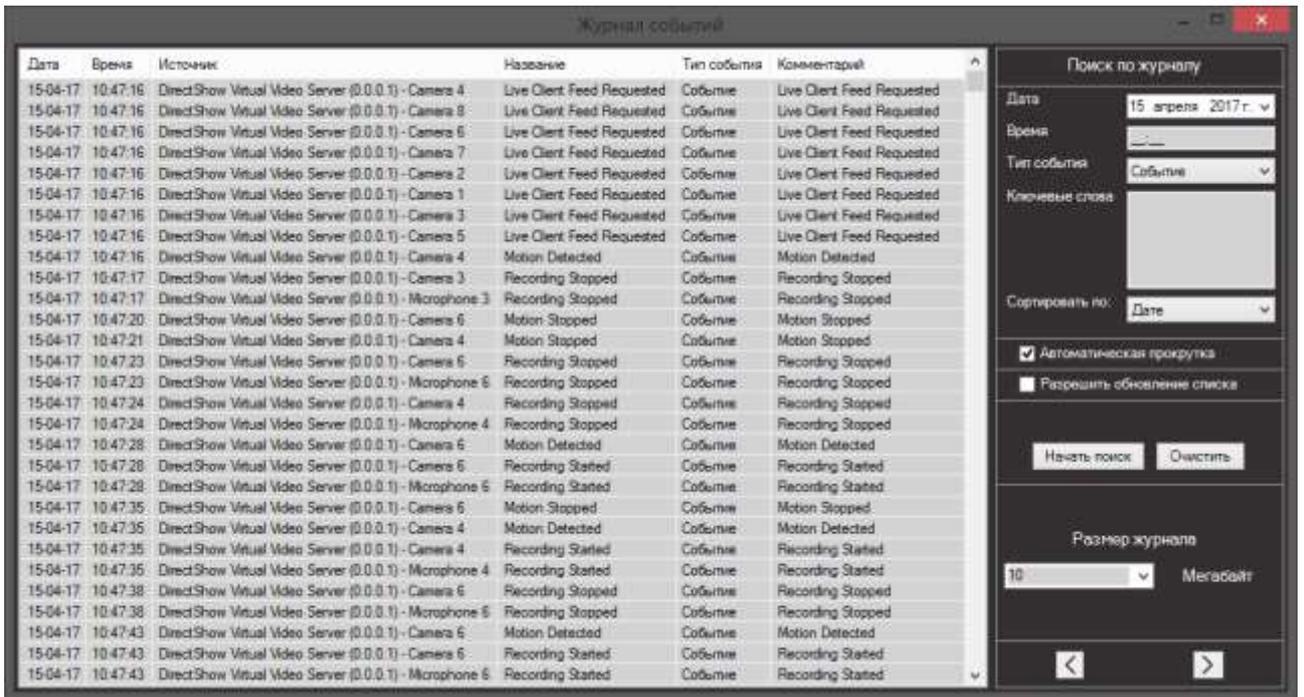


Рисунок Б7 – Журнал событий компонента AM.Eye – Big

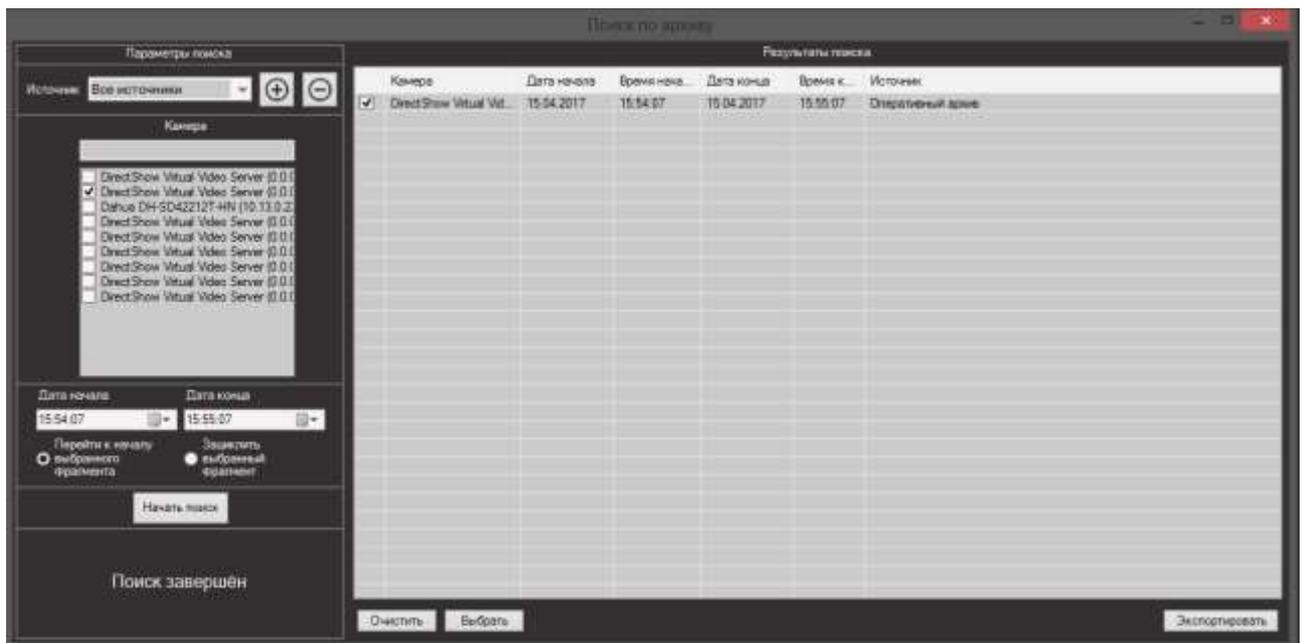


Рисунок Б8 – Поиск по архиву компонента AM.Eye – Big

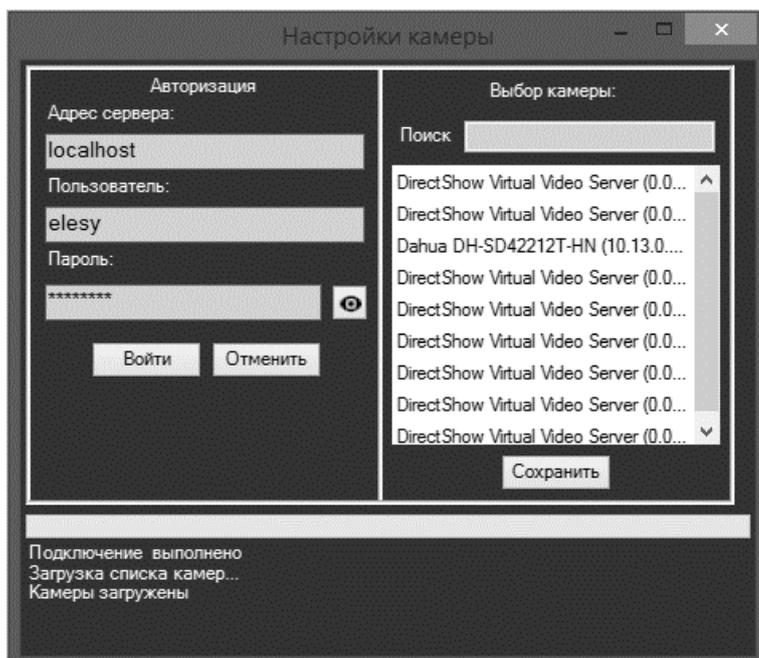


Рисунок Б9 – Настройки с дополнительными функциями компонента AM.Eye – Mini



Рисунок Б10 – Состояние встроенного рабочего компонента при больших размерах компонента AM.Eye – Mini



Рисунок Б11 – Отображение видеопотока в отдельном окне с дополнительными функциями компонента AM.Eye – Mini

Частичный листинг подсистемы AM.Traffic

Файл AMTraffic.cs

```

using System;
using System.ServiceProcess;
using System.Collections.Generic;
using System.Threading.Tasks;
using NLog;
using System.Diagnostics;
using System.Reflection;

///Оpc пространства имен (Opc.Ua, Opc.Ua.Configuration - OPCFoundation)
using Opc;
using Opc.Ua;
using Opc.Ua.Configuration;
using AM.SentinelHasp.Licenses;
using AM.SentinelHasp.Vendors;
using AM.SentinelHasp.Factories;

namespace Client
{
    public class AmTraffic : ServiceBase
    {
        private static OpcServer server;
        private static RabbitMQClient client;
        private static Logger log = LogManager.GetCurrentClassLogger();
        private static MessageData.MessageData data = new
MessageData.MessageData();
        private static bool isServerStarted = false;
        private static Decimal messageNumder = 0;

        #region LicenseData
        private static int id = 1; //Идентификатор
продукта для HASP
        private static int[] code = new int[] { 40021 }; //Код для HASP
        private static string encode =
"86X6GPFW32DdxGIJUd6uTjgI0dupyD3UI2xR3jJngynLXBW2l2mdE0fdA49glxnRr7w=";
//Зашифрованная строка на основании Id и features для HASP
        private static int timeCheckLicense = 1;
        private static Status lastLicenseStatus = Status.Unknow;
        #endregion

        /// <summary>
        /// Запуск службы
        /// </summary>
        protected override void OnStart(string[] args)
        {
            try
            {
                Debugger.Launch();
                StartOpcUaServer();
                //CheckLicence();

                Task.Run(() =>
                {
                    try
                    {

```

```

        RabbitMQClient.MessageReceived +=
RabbitMQClient_MessageReceived;
        client = new RabbitMQClient();
    }
    catch (Exception err)
    {
        log.Fatal(err, $"Фатальная ошибка: {err.Message}");
        SetDiagnosticMessage("Фатальная ошибка при работе
RabbitMQClient. Для получения дополнительной информации обратитесь к лог-файлу",
true);
    }

    });
    log.Info("Служба AM.Traffic запущена");
    SetDiagnosticMessage("Сервер запущен успешно", false);
}
catch (Exception)
{
    Environment.Exit(0);
}
}

/// <summary>
/// Остановка службы
/// </summary>
protected override void OnStop()
{
    log.Info("Служба AM.Traffic остановлена");
}

/// <summary>
/// Обновление диагностических сообщений
/// </summary>
internal static void SetDiagnosticMessage(string message, bool isError)
{
    if(isServerStarted)
    {
        data.DiagnosticMessage = message;
        data.IsSomethingWrong = isError;
        try
        {
            server.GetNodeManager().UpdateData(data);
        }
        catch (Exception err)
        {
            log.Error("Невозможно установить диагностическое сообщение",
err.Message);
        }
    }
}

/// <summary>
/// Получение сообщения от RabbitMQ клиента
/// </summary>
private static bool RabbitMQClient_MessageReceived(string message)
{
    log.Info("Получено новое сообщение: " + message);
    if(server != null)
    {
        try
        {
            data = MessageParser.MessageParser.JsonToModel(message);
            server.GetNodeManager().UpdateData(data);
        }
    }
}

```

```

        SetDiagnosticMessage($"Данные сообщения
\"){++messageNumber}\" успешно обработаны", false);
        return true;
    }
    catch (Exception err)
    {
        log.Error(err, "Ошибка при обновлении данных: " +
err.Message);
        //TODO: При ошибке обновления данных необходимо не
подтверждать получение сообщения, временное решение для фильтрации инцидентов
        //SetDiagnosticMessage("Ошибка при обновлении данных",
true);
        return true;
    }
    }
    return false;
}

/// <summary>
/// Создание и запуск OPC UA сервера
/// </summary>
private static void StartOpcUaServer()
{
    server = new OpcServer("AMTraffic");
    ApplicationInstance application = new ApplicationInstance();
    application.ApplicationType = ApplicationType.Server;
    application.ConfigSectionName =
FileVersionInfo.GetVersionInfo(Assembly.GetExecutingAssembly().Location).Product
Name;
    ApplicationConfiguration config = null;

    //Загрузка конфигурации
    try
    {
        config = application.LoadApplicationConfiguration(false);
    }
    catch (Exception err)
    {
        log.Error("AmTraffic.StartOpcUaServer Невозможно загрузить
конфигурацию OPC сервера: " + err.Message);
        Environment.Exit(0);
    }

    //Попытка проверить сертификат
    try
    {
        application.CheckApplicationInstanceCertificate(false, 0);
    }
    catch (Exception err)
    {
        log.Error("AmTraffic.StartOpcUaServer Невозможно проверить
сертификат OPC сервера: " + err.Message);
        Environment.Exit(0);
    }

    //Попытка запустить сервер
    try
    {
        application.Start(server);
    }
    catch (Exception err)
    {
        log.Error("AmTraffic.StartOpcUaServer ошибка при запуске
сервера: " + err.Message);
    }
}

```

```

        Environment.Exit(0);
    }

    //Инициализация дерева
    try
    {
        data.DiagnosticMessage = "Запуск сервера...";
        data.IsSomethingWrong = false;

server.GetNodeManager().ConvertDataToOpc(typeof(MessageData.MessageData), data,
"AMTraffic");
    }
    catch (Exception err)
    {
        log.Error("Ошибка при создании Оpc дерева " + err.Message);
        Environment.Exit(0);
    }

    //Сервер успешно запущен
    isServerStarted = true;
    AmTraffic.SetDiagnosticMessage("Сервер успешно запущен", false);
}

/// <summary>
/// Проверка лицензии HASP
/// </summary>
private static void CheckLicence()
{
    var vendorInfo = new VendorInfo(id, VendorCodes.KDCYB, encode);
    var licenseManager = new LicenseManager();
    licenseManager.OnAfterCheking += License_OnChecking;
    try
    {
        licenseManager.Check(ServiceFactory.Create(), vendorInfo, code,
timeCheckLicense);
    }
    catch (Exception err)
    {
        log.Error("Ошибка при проверке ключа HASP");
        SetDiagnosticMessage("Ошибка при проверке ключа HASP", true);
        Environment.Exit(0);
    }
}

/// <summary>
/// Сообщение о состоянии проверки лицензии
/// </summary>
private static void License_OnChecking(License license, Status status,
string message)
{
    if (status == Status.Ok && status != lastLicenseStatus)
    {
        lastLicenseStatus = Status.Ok;
        SetDiagnosticMessage("Лицензионный ключ успешно найден", false);
        log.Info($"Лицензия: staus: {license.ActualStatus}, type:
{license.Type}. Успешно найдена!");
    }
    else if (status == Status.FeatureExpired && status !=
lastLicenseStatus)
    {
        lastLicenseStatus = status;
        SetDiagnosticMessage("Лицензия недействительна", true);
        log.Error("Лицензия недействительна");
    }
}

```

```

        Environment.Exit(0);
    }
    else if(status != lastLicenseStatus)
    {
        lastLicenseStatus = status;
        SetDiagnosticMessage($"Не удалось найти лицензию", false);
        log.Error($"Не удалось найти лицензию: {message}");
        Environment.Exit(0);
    }
}

private void InitializeComponent()
{
    this.ServiceName =
FileVersionInfo.GetVersionInfo(Assembly.GetExecutingAssembly().Location).Product
Name;
}
}
}

```

Файл RabbitMQClient.cs

```

using System;
using System.Text;
using System.Threading.Tasks;
using RabbitMQ.Client;
using NLog;
using System.IO;
using RabbitMQ.Client.Events;
using System.Threading;
using System.Diagnostics;
using RabbitMQ.Client.MessagePatterns;

namespace Client
{
    public class RabbitMQClient
    {
        private static Logger log;
        public delegate bool OnMessageReceived(string message);
        public delegate void OnCheckMessage(IModel channel, string queueName);
        public static event OnMessageReceived MessageReceived;

        public RabbitMQClient()
        {
            log = LogManager.GetCurrentClassLogger();
            //SettingsData.GeneralSettingsData generalData;
            SettingsData.RabbitMQSettingsData rabbitData;
            ConnectionFactory factory = new ConnectionFactory();
            try
            {
                //Сохранение настроек
                if (!File.Exists(JsonSettings.Settings.FilePath))
                {
                    SettingsData.RabbitMQSettingsData defaultRabbitData = new
SettingsData.RabbitMQSettingsData();

                    JsonSettings.Settings.SaveSettings(SettingsData.RabbitMQSettingsData.InstanceName, defaultRabbitData);
                }
            }
        }
    }
}

```

```

        rabbitData =
        (SettingsData.RabbitMQSettingsData) JsonSettings.Settings.GetSettings<SettingsData.RabbitMQSettingsData>(SettingsData.RabbitMQSettingsData.InstanceName,
        typeof(SettingsData.RabbitMQSettingsData));
        factory.UserName = rabbitData.UserName;
        factory.Password = rabbitData.Password;
        factory.HostName = rabbitData.HostName;
        factory.VirtualHost = rabbitData.VirtualHost;

        Task.Run(() =>
        {
            while (true)//Цикл для пересоздания подключения в случае
ошибки
            {
                try
                {
                    factory.Protocol = Protocols.AMQP_0_9_1;

                    using (var connection = factory.CreateConnection())
                    {
                        using (var channel = connection.CreateModel())
                        {
                            channel.ExchangeDeclare(rabbitData.ExchangeName, rabbitData.ExchangeType,
                            rabbitData.ExchangeDurable, rabbitData.ExchangeAutodelete);
                            channel.QueueDeclare(rabbitData.QueueName,
                            rabbitData.QueueDurable, rabbitData.QueueExclusive, rabbitData.QueueAutodelete,
                            null);
                            channel.QueueBind(rabbitData.QueueName,
                            rabbitData.ExchangeName, rabbitData.RoutingKey, null);

                            //Метод push/subscribe
                            var subscription = new Subscription(channel,
                            rabbitData.QueueName, false);

                            while (true)
                            {
                                BasicDeliverEventArgs
                                basicDeliveryEventArgs = subscription.Next();
                                string message =
                                Encoding.UTF8.GetString(basicDeliveryEventArgs.Body);
                                bool success = false;
                                if (MessageReceived != null)
                                {
                                    success = MessageReceived(message);
                                }
                                else
                                {
                                    log.Warn("RabbitMQClient.RunCheckingMessages Никто не подписан на получение
                                    сообщения. Сообщение не обработано");

                                    AmTraffic.SetDiagnosticMessage("Сообщение не было получено, однако не было
                                    обработано", true);

                                    break;
                                }
                            }
                            try
                            {
                                //Если в процессе обработки все
                                прошло нормально

                                if (success)
                                {

```


Частичный листинг подсистемы AM.Eye

Файл MiniEye.cs

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Drawing;
using Microsoft.Win32;
using System.Reflection;
using System.Text;
using System.Windows.Forms;
using System.Runtime.InteropServices;
using Settings;
using Contract;
using System.Diagnostics;
using Contract.Auth;
using System.Threading.Tasks;
using Ole.Controls;
using AM.SentinelHasp.Licenses;
using AM.SentinelHasp.Vendors;
using AM.SentinelHasp.Factories;

namespace MiniEye
{
    [Guid("9E86F8DE-0BF3-4762-B936-33D0A8C2BC01")]
    [ClassInterface(ClassInterfaceType.None)]
    [ComVisible(true)]
    public partial class MiniEye : ActiveXControlBase
    {
        #region LicenseData
        private static int id = 1; //Идентификатор
        продукта для HASP
        private static int[] code = new int[] { 40018 }; //Код для HASP
        private static string encode =
        "/WLxUN5Ukr9PMer1U2T8RvMaGxna3QysKTDF8QNDFt1nRqpusThnc3QX/1rMUU51AWg=";
        //Зашифрованная строка на основании Id и features для HASP
        private static int timeCheckLicense = 1;
        private static Status lastLicenseStatus = Status.Unknow;
        #endregion

        #region Register ActiveX
        /// <summary>
        /// Регистрация компонента ActiveX
        /// </summary>
        [ComRegisterFunction()]
        public static void RegisterCOM(string key)
        {
            // Strip off HKEY_CLASSES_ROOT\ from the passed key as I don't need
            it
            StringBuilder sb = new StringBuilder(key);

            sb.Replace(@"HKEY_CLASSES_ROOT\", "");
            // Open the CLSID\{guid} key for write access
            RegistryKey k = Registry.ClassesRoot.OpenSubKey(sb.ToString(),
            true);

            //Set beautiful name for object
            k.SetValue("", MiniEye._FriendlyName);

```

```

in
    // And create the 'Control' key - this allows it to show up
    // the ActiveX control container
    RegistryKey ctrl = k.CreateSubKey("Control");
    ctrl.Close();
    RegistryKey insertb = k.CreateSubKey("Insertable");
    insertb.Close();

    // Next create the CodeBase entry - needed if not string named
and GACced.
    RegistryKey inprocServer32 = k.OpenSubKey("InprocServer32", true);
    inprocServer32.SetValue("CodeBase",
Assembly.GetExecutingAssembly().CodeBase);
    inprocServer32.Close();
    // Finally close the main key
    k.Close();
}

/// <summary>
/// Отмена регистрации компонента ActiveX
/// </summary>
/// <param name="key">The registry key</param>
[ComUnregisterFunction()]
public static void Unregister(string key)
{
    StringBuilder sb = new StringBuilder(key);
    sb.Replace(@"HKEY_CLASSES_ROOT\", "");

    // Open HKCR\CLSID\{guid} for write access
    RegistryKey k = Registry.ClassesRoot.OpenSubKey(sb.ToString(),
true);

    // Delete the 'Control' key, but don't throw an exception if it
does not exist
    k.DeleteSubKey("Control", false);

    k.DeleteSubKey("Insertable", false);

    // Next open up InprocServer32
    //RegistryKey inprocServer32 =
k.OpenSubKey("InprocServer32", true);

    // And delete the CodeBase key, again not throwing if missing
k.DeleteSubKey("CodeBase", false);

    // Finally close the main key
    k.Close();
}
#endregion

#region Properties
private const string _FriendlyName = "AM.Eye - Mini";
private bool _PreviewHasBeenShown = false;
private bool _CameraSwitching = false;
private bool _IsConnected = false;
private new bool IsDisposed = false;

//Объекты отображения
private Views.Preview _ViewPreview = null;
private Views.Settings _ViewSettings = null;
private Views.SmallPreview _SmallPreview = null;
private Contract.IVideoModel _VideoLive = null;

```

```

#region Save state properties
[Browsable(false)]
//[Description(""), Category("Data")]
public bool IsStateSaved { get; set; }
public string ServerName { get; set; }
public AuthType AuthType { get; set; }
public string UserName { get; set; }
public string Password { get; set; }
//Свойства могут быть сохранены только в виде примитивных типов, поэтому
камера
//конвертируется в строку
public string SerializedCamera { get; set; }
#endregion Save state properties

#endregion Properties

/// <summary>
/// Создает все объекты.
/// Восстановление состояния объекта происходит после вызова
конструктора
/// основная работа начинается в OnControlStartRunTime, когда из режима
разработки HMI
/// производится переход в режим исполнения
/// </summary>
public MiniEye()
{
    InitializeComponent();
    this.Dock = DockStyle.Fill;
    this.AutoSize = true;
    this.Size = new System.Drawing.Size(0, 0);
    _VideoLive = ServerManager.GetVideoLive();
}

/// <summary>
/// Выполняет очистку используемых ресурсов
/// </summary>
private void MiniEye_HandleDestroyed(object sender, EventArgs e)
{
    ServerManager.Dispose();
    _VideoLive?.Disconnect();
    _VideoLive?.Dispose();
    _ViewPreview?.Dispose();
    _ViewSettings?.Dispose();
    _SmallPreview?.Dispose();
    _ViewPreview = null;
    _ViewSettings = null;
    _SmallPreview = null;
    _VideoLive = null;
}

#region Override methods
protected override void OnControlStopRunTime()
{
    this.Controls.Clear();
    IsDisposed = true;
    if (_SmallPreview != null && !_SmallPreview.IsDisposed)
        _SmallPreview.GetVideoSpace().Click -= SmallPreviewPanel_Click;
    ImageAnimator.StopAnimate(this.BackgroundImage, OnFrameChanged);
    this.MouseClick -= MiniEye_MouseClick;
    this.SizeChanged -= MiniEye_SizeChanged;
    this.HandleDestroyed -= MiniEye_HandleDestroyed;
    MiniEye_HandleDestroyed(null, null);
}

```

```

        this.BackColor = Color.DarkGray;
        this.BackgroundImage = null;
    }

private void LicenseNotFound()
{
    this.Invoke(new MethodInvoker(() => {
        OnControlStopRunTime();
        Label errorText = new Label();
        errorText.Text = "AM.Eye - Mini. Лицензия не обнаружена";
        errorText.AutoSize = true;
        errorText.ForeColor = Color.Black;
        this.Controls.Clear();
        this.Controls.Add(errorText);
    }));
}

protected override void OnControlStartRunTime()
{
    //Проверка лицензии перед запуском
    try
    {
        //CheckLicence();
        lastLicenseStatus = Status.Ok;
    }
    catch (Exception err)
    {
        Trace.TraceError($"Sentiel key error: {err}");
        LicenseNotFound();
        return;
    }
    if (lastLicenseStatus == Status.Ok)
    {
        //Приостановить проверку лицензии
        licenseManager.OnAfterCheking -= License_OnChecking;
        this.Controls.Clear();
        IsDisposed = false;
        InitializeComponent();
        SetStyle(ControlStyles.SupportsTransparentBackColor, true);
        this.BackColor = Color.DarkGray;
        this.Dock = DockStyle.Fill;
        _VideoLive = ServerManager.GetVideoLive();
        //Проверка размеров иконки, если у иконки размеры больше 200*200
то показывается камера
        this.HandleDestroyed += MiniEye_HandleDestroyed;
        this.MouseClick += MiniEye_MouseClick;
        Connect();
        this.SizeChanged += MiniEye_SizeChanged;
        //Возобновить проверку лицензии
        licenseManager.OnAfterCheking += License_OnChecking;
    }
    else
    {
        LicenseNotFound();
        return;
    }
}
#endregion

#region Settings Handlers

/// <summary>
/// Обработчики событий настроек
/// </summary>

```

```

private void ConnectSettingsHandlers()
{
    _ViewSettings.OnCheckConnection += _ViewSettings_OnCheckConnection;
    _ViewSettings.OnGetCameraRequest +=
_ViewSettings_OnGetCameraRequest;
    _ViewSettings.OnCameraNameChanged +=
_ViewSettings_OnCameraNameChanged;
    _ViewSettings.OnCameraSelected += _ViewSettings_OnCameraSelected;
}

/// <summary>
/// Обработчик переименования имени камеры
/// </summary>
private void _ViewSettings_OnCameraNameChanged(List<ICameraModel>
cameras)
{
    CameraConfigurationManager.Instance.Save(cameras);
}

/// <summary>
/// Получение списка камер
/// </summary>
private List<ICameraModel> _ViewSettings_OnGetCameraRequest()
{
    List<ICameraModel> listCamera =
CameraConfigurationManager.Instance.CameraLiveList;
    if (listCamera.Count < 1) throw new Exception("Невозможно получить
камеры с сервера");
    return listCamera;
}

/// <summary>
/// Проверка соединения с видео сервером
/// </summary>
///
private static object LockObject = new object();
private static void _ViewSettings_OnCheckConnection(string server,
AuthType authType, string login, string password)
{
    lock (LockObject)
    {
        if (server.Contains(", "))
            throw new Exception("Не удастся найти сервер. Проверьте
адрес подключения");
        //TODO: Добавить тип авторизации при необходимости
        AuthData authData = new AuthData(server, login, password);
        ServerManager.Instance._ModelConnection.Connect(authData);
        switch (ServerManager.Instance._ModelConnection.Status)
        {
            case ConnectStatus.Ok:
                break;
            case ConnectStatus.ServerNotFound:
                throw new Exception("Не удастся найти сервер. Проверьте
адрес подключения");
            case ConnectStatus.IncorrectPassOrLogin:
                throw new Exception("Логин/пароль введены неверно");
            default:
                throw new Exception("Неизвестная ошибка при
подключении");
        }
    }
}

/// <summary>

```

```

/// Показать изображение после выбора камеры в настройках
/// </summary>
private void _ViewSettings_OnCameraSelected(LoginAndCameraData data)
{
    _CameraSwitching = true;
    SetData(data);
    _ViewSettings.Close();
    if (_PreviewHasBeenShown)
        InitializePreview();
    _IsConnected = true;
    StateChanging();//После выбора камеры проверить куда выводить
изображение в новое окно или в область значка
}

#endregion Settings Handlers

#region Preview Handlers

/// <summary>
/// Обработчики событий preview
/// </summary>
private void ConnectPreviewHandlers()
{
    _ViewPreview.OnSettingsChange += _ViewPreview_OnSettingsChange;
    _ViewPreview.OnClose += _ViewPreview_OnClose;
}

private void _ViewPreview_OnClose()
{
    _PreviewHasBeenShown = false;
}

private void _ViewPreview_OnSettingsChange()
{
    _ViewPreview.Close();
    InitializeSettings();
    _ViewSettings.Show();
}

#endregion Preview Handlers

#region SmallPreview Handlers
private void ConnectSmallPreviewHandlers()
{
    _SmallPreview.GetVideoSpace().Click += SmallPreviewPanel_Click;
}

private void SmallPreviewPanel_Click(object sender, EventArgs e)
{
    contextMenuStrip.Show(Cursor.Position);
}
#endregion SmallPreview Handlers

/// <summary>
/// Запустить компонент
/// </summary>
private void cameraButton_Click(object sender, EventArgs e)
{
    contextMenuStrip.Show(Cursor.Position);
}

/// <summary>
/// При изменении размеров проверить состояние объекта

```

```

/// </summary>
private void MiniEye_SizeChanged(object sender, EventArgs e)
{
    StateChanging();
}

/// <summary>
/// Необходимо вызвать при изменении состояния объекта
/// </summary>
private void StateChanging()
{
    //Без подключения дальше ничего не делается
    if (!_IsConnected)
        return;
    else
        reconnectToolStripMenuItem.Enabled = false;

    //Выключить возможность повторного подключения
    reconnectToolStripMenuItem.Enabled = false;
    //Открыть доп. меню просмотра в окне
    if (IsStateSaved)
        showInTheWindowToolStripMenuItem.Enabled = true;
    else
        showInTheWindowToolStripMenuItem.Enabled = false;

    //Состояние в зависимости от размеров
    if (this.Width > 200 && this.Height > 200)
    {
        if (!IsStateSaved)
        {
            ImageAnimator.StopAnimate(this.BackgroundImage,
OnFrameChanged);

            this.Controls.Clear();
            //Обработчик для анимации backgroundImage
            this.BackgroundImage = Properties.Resources.loading128;
            ImageAnimator.Animate(this.BackgroundImage, OnFrameChanged);
            this.BackgroundImageLayout = ImageLayout.Stretch;
        }
        else
        {
            if (this.Controls.Count == 0 || !(this.Controls[0] is
Views.SmallPreview) || _CameraSwitching) //Избежать лишних переподключений и
обновлений
            {
                this.BackgroundImage = null;
                InitializeSmallPreview(); //Встраивается панель видео
                _CameraSwitching = false;
            }
        }
    }
    else if (this.Width < 200 || this.Height < 200)
    {
        //TODO: очистка ресурсов?
        this.Controls.Clear();
        this.BackgroundImage = Properties.Resources.Camera256;
    }
}

/// <summary>
/// Инициализация окна предварительного просмотра изображения
/// также сопоставление окна с камерой
/// </summary>

```

```

private void InitializePreview()
{
    if (_ViewPreview == null || _ViewPreview.IsDisposed)
        _ViewPreview = new Views.Preview(GetData());
    //Привязка отображения к камере
    try
    {
        //Нужно сначала показать пользователю форму, после чего
вставлять в нее видео
        _ViewPreview.Show();

        _ViewPreview.SetCamera(ServerManager.Instance._ModelSerializeDevice.Deserialize(
GetData().Camera));
        ConnectPreviewHandlers();
    }
    catch (Exception)
    {
        Debug.WriteLine("Ошибка инициализации Preview");
        //Рекурсия
        InitializePreview();
        this.StateChanging();
    }
}

private void InitializeSmallPreview()
{
    _SmallPreview = new Views.SmallPreview();
    _SmallPreview.Width = this.Width;
    _SmallPreview.Height = this.Height;
    _SmallPreview.Dock = DockStyle.Fill;
    ImageAnimator.StopAnimate(this.BackgroundImage, OnFrameChanged);
    this.BackgroundImage = null;
    this.Controls.Clear();
    this.Controls.Add(_SmallPreview);
    ConnectSmallPreviewHandlers();
    _SmallPreview.Show();
    _SmallPreview.Paint += _SmallPreview_Paint;
    _SmallPreview.Refresh();
    _SmallPreview.GetVideoSpace().BeginInvoke((Action)delegate
    {
        try
        {
            _VideoLive.SetVideoStreamInPanel(ServerManager.Instance._ModelSerializeDevice.De
serialize(SerializedCamera), _SmallPreview.GetVideoSpace());
        }
        catch
        {
            Debug.WriteLine("Ошибка инициализации SmallPreview");
        }
    });
}

/// <summary>
/// Хак для Milestone, чтобы при задержке отрисовки UI 100%
/// показан был компонент перед инициализацией
/// </summary>
private void _SmallPreview_Paint(object sender, PaintEventArgs e)
{
    try
    {
        _SmallPreview.Paint -= _SmallPreview_Paint;
    }
}

```

```

_VideoLive.SetVideoStreamInPanel (ServerManager.Instance._ModelSerializeDevice.De
serialize (SerializedCamera), _SmallPreview.GetVideoSpace());
        _SmallPreview.Refresh();
    }
    catch (Exception err)
    {
        _SmallPreview.Paint += _SmallPreview_Paint;
        Debug.WriteLine(err.Message);
    }
}

/// <summary>
/// Инициализация окна настроек
/// </summary>
private void InitializeSettings()
{
    if (_ViewSettings == null || _ViewSettings.IsDisposed)
    {
        _ViewSettings = new Views.Settings (GetData());
        ConnectSettingsHandlers();
    }
}

/// <summary>
/// Записать данные для сохранения
/// </summary>
/// <param name="data"></param>
private void SetData(LoginAndCameraData data)
{
    this.IsStateSaved = true; //Сохранить состояние контрола
    this.UserName = data.UserName;
    this.Password = data.Password;
    this.ServerName = data.ServerName;
    this.AuthType = AuthType.Basic;
    this.SerializedCamera = null;
    this.SerializedCamera = data.Camera;
    data.saveData();
    StateChanging();
}

/// <summary>
/// Формирует объект данных и возвращает его
/// </summary>
private LoginAndCameraData GetData()
{
    LoginAndCameraData data;
    if (IsStateSaved)
        return new LoginAndCameraData(this.Password, this.ServerName,
this.UserName, this.SerializedCamera);

    try
    {
        data =
Settings.SettingManager.GetConfig<LoginAndCameraData>(LoginAndCameraData.CONFIG_
KEY);
        if (data == null) throw new Exception("Нет сохраненных данных");
    }
    catch (Exception)
    {
        data = new LoginAndCameraData();
    }
    return data;
}
}

```

```

/// <summary>
/// Анимация фонового изображения кнопки в отдельном потоке
/// </summary>
private void OnFrameChanged(object sender, EventArgs e)
{
    if (InvokeRequired)
    {
        BeginInvoke((Action)(() => OnFrameChanged(sender, e)));
        return;
    }
    ImageAnimator.UpdateFrames();
    Invalidate(false);
}

private void MiniEye_MouseClick(object sender, MouseEventArgs e)
{
    this.contextMenuStrip.Show(Cursor.Position);
}

/// <summary>
/// Настройка компонента из контекстного меню
/// </summary>
private void SetupToolStripMenuItem_Click(object sender, EventArgs e)
{
    this.InitializeSettings();
    _ViewSettings.Show();
}

/// <summary>
/// Просмотр в отдельном окне из контекстного меню
/// </summary>
/// <param name="sender"></param>
/// <param name="e"></param>
private void ShowInTheWindowToolStripMenuItem_Click(object sender,
EventArgs e)
{
    this.InitializePreview();
}

/// <summary>
/// Повторное подключение из контекстного меню
/// </summary>
private void ReconnectToolStripMenuItem_Click(object sender, EventArgs
e)
{
    Connect();
}

private void MiniEye_Load(object sender, EventArgs e)
{
    Connect();
}

private void Connect()
{
    //Не выполнять автоматическое подключение если состояние не
сохранено
    if (!IsStateSaved)
    {
        this.BackgroundImage = Properties.Resources.Settings256;
        this.contextMenuStrip.Enabled = true;
        this.setupToolStripMenuItem.Enabled = true;
        return;
    }
}

```

```

    }

    this.contextMenuStrip.Enabled = false;
    this.BackgroundImage = Properties.Resources.loading128;
    ImageAnimator.Animate(this.BackgroundImage, OnFrameChanged);
    //Подключиться и загрузить камеры
    //Запустить авторизацию и получение списка камер в отдельном потоке
    Task.Run(() =>
    {
        try
        {
            //Подключение к серверу
            LoginAndCameraData data = GetData();
            _ViewSettings_OnCheckConnection(data.ServerName,
            Contract.Auth.AuthType.Basic, data.UserName, data.Password);
            //Запрос списка камер
            List<ICameraModel> cameras =
            _ViewSettings_OnGetCameraRequest();
            try
            {
                if
                (ServerManager.Instance._ModelCameraManager.GetCamera(ServerManager.Instance._ModelSerializeDevice.Deserialize(data.Camera).Id) == null)
                    throw new Exception("Нет камер для восстановления");
            }
            catch (Exception)
            {
                throw new Exception("Ошибка восстановления камеры");
            }

            this.Invoke((MethodInvoker)delegate
            {
                if (!this.IsDisposed)
                {
                    ImageAnimator.StopAnimate(this.BackgroundImage,
                    OnFrameChanged);

                    _IsConnected = true;
                    this.BackgroundImage =
                    Properties.Resources.Camera256;
                    //настройки контекстного меню
                    this.contextMenuStrip.Enabled = true;
                    this.reconnectToolStripMenuItem.Enabled = false;
                    this.setupToolStripMenuItem.Enabled = true;
                    this.showInTheWindowToolStripMenuItem.Enabled =
                    false;

                    StateChanging();
                }
            });
        }
        catch (Exception)
        {
            this.Invoke((MethodInvoker)delegate
            {
                if (!this.IsDisposed)
                {
                    ImageAnimator.StopAnimate(this.BackgroundImage,
                    OnFrameChanged);

                    //На случай если камеры не загрузились
                    _IsConnected = false;
                    this.BackgroundImage =
                    Properties.Resources.Settings256;
                    //настройки контекстного меню
                    this.contextMenuStrip.Enabled = true;
                }
            });
        }
    });
}

```

