

Школа информационных технологий и робототехники
 Направление подготовки 09.03.02 «Информационные системы и технологии»
 Отделение школы (НОЦ) информационных технологий

БАКАЛАВРСКАЯ РАБОТА

Тема работы
Серверное программное обеспечение и веб-приложение для информационной системы с технологией интерактивной визуализации средствами дополненной реальности

УДК 004.774:004.455.1'236:004.93

Студент

Группа	ФИО	Подпись	Дата
8И6А	Степаненко Владимир Владимирович		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Цапко С.Г.	к.т.н.		

КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН	Подопригора И.В.	к.э.н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент ООД	Немцова О.А.			

ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Цапко И.В.	к.т.н.		

Результаты обучения

Код результатов	Результат обучения (выпускник должен быть готов)
<i>Профессиональные и общепрофессиональные компетенции</i>	
P1	Применять базовые и специальные естественнонаучные и математические знания для комплексной инженерной деятельности по созданию, внедрению и эксплуатации геоинформационных систем и технологий, а также информационных систем и технологий в бизнесе.
P2	Применять базовые и специальные знания в области современных информационных технологий для решения инженерных задач.
P3	Ставить и решать задачи комплексного анализа, связанные с созданием геоинформационных систем и технологий, информационных систем в бизнесе, с использованием базовых и специальных знаний, современных аналитических методов и моделей.
P4	Выполнять комплексные инженерные проекты по созданию информационных систем и технологий, а также средств их реализации (информационных, методических, математических, алгоритмических, технических и программных).
P5	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 геоинформационных систем и технологий, а также информационных систем и технологий в бизнесе.
P6	Внедрять, эксплуатировать и обслуживать современные геоинформационные системы и технологии, информационные системы и технологии в бизнесе, обеспечивать их высокую эффективность, соблюдать правила охраны здоровья, безопасность труда, выполнять требования по защите окружающей среды.
<i>Универсальные (общекультурные) компетенции</i>	
P7	Использовать базовые и специальные знания в области проектного менеджмента для ведения комплексной инженерной деятельности.
P8	Осуществлять коммуникации в профессиональной среде и в обществе в целом. Владеть иностранным языком (углублённый английский язык), позволяющем работать в иноязычной среде, разрабатывать документацию, презентовать и защищать результаты комплексной инженерной деятельности.
P9	Эффективно работать индивидуально и в качестве члена команды, состоящей из специалистов различных направлений и квалификаций,
P10	Демонстрировать личную ответственность за результаты работы и готовность следовать профессиональной этике и нормам ведения комплексной инженерной деятельности.
P11	Демонстрировать знания правовых, социальных, экологических и культурных аспектов комплексной инженерной деятельности, а также готовность к достижению должного уровня физической подготовленности для обеспечения полноценной социальной и профессиональной деятельности.

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа информационных технологий и робототехники
Направление подготовки 09.03.02 «Информационные системы и технологии»
Отделение школы (НОЦ) информационных технологий

УТВЕРЖДАЮ:
Руководитель ООП

(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
8И6А	Степаненко Владимиру Владимировичу

Тема работы:

Серверное программное обеспечение и веб-приложение для информационной системы с технологией интерактивной визуализации средствами дополненной реальности	
Утверждена приказом директора (дата, номер)	№154-52/с от 02.06.2020

Срок сдачи студентом выполненной работы:	06.06.2020
--	------------

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

Исходные данные к работе	Разработка серверного программного обеспечения и веб-приложения для информационной системы с технологией интерактивной визуализации средствами дополненной реальности
---------------------------------	---

Перечень подлежащих исследованию, проектированию и разработке вопросов	Определение требований к серверному компоненту, определение требований к веб-клиенту, выбор стека технологий, проектирование БД, разработка серверного компонента, проектирование пользовательского интерфейса, разработка клиентского веб-приложения.
Перечень графического материала	Презентация в формате *.pptx
Консультанты по разделам выпускной квалификационной работы (с указанием разделов)	
Раздел	Консультант
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Подопригора Игнат Валерьевич
Социальная ответственность	Немцова Ольга Александровна
Названия разделов, которые должны быть написаны на русском и иностранном языках:	
Заключение	

Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	14.01.2020
---	------------

Задание выдал руководитель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Цапко С.Г.	к.т.н		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8И6А	Степаненко Владимир Владимирович		

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа информационных технологий и робототехники
 Направление подготовки 09.03.02 «Информационные системы и технологии»
 Уровень образования бакалавриат
 Отделение школы (НОЦ) информационных технологий
 Период выполнения весенний семестр 2019/2020 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:	06.06.2020
--	------------

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
02.06.2020	Основная часть	75
04.05.2020	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	15
04.05.2020	Социальная ответственность	10

СОСТАВИЛ:

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Цапко С.Г.	к.т.н.		

СОГЛАСОВАНО:

Руководитель ООП

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Цапко И.В.	к.т.н.		

ЗАДАНИЕ ДЛЯ РАЗДЕЛА «ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ»

Студенту:

Группа	ФИО
8И6А	Степаненко Владимиру Владимировичу

Школа	ИШИТР	Отделение школы (НОЦ)	ОИТ
Уровень образования	Бакалавриат	Направление/специальность	09.03.02 Информационные системы и технологии

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

1. Стоимость ресурсов научного исследования (НИИ): материально-технических, энергетических, финансовых, информационных и человеческих	Оклад инженера – 21760 руб. Оклад руководителя – 33664 руб.
2. Нормы и нормативы расходования ресурсов	Премияльный коэффициент 30%; Коэффициент доплат и надбавок 20%; Районный коэффициент 30%; Коэффициент дополнительной заработной платы 12%; Накладные расходы 16%.
3. Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования	Коэффициент отчислений на уплату во внебюджетные фонды 30%

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. Оценка коммерческого потенциала, перспективности и альтернатив проведения НИИ с позиции ресурсоэффективности и ресурсосбережения	-Анализ конкурентных технических решений
2. Планирование и формирование бюджета научных исследований	Формирование плана и графика разработки: - определение структуры и трудоемкости работ; - разработка графика Ганта. Формирование бюджета затрат: - материальные затраты; - затраты на специальное оборудование; - заработная плата (основная и дополнительная); - отчисления на социальные цели; - накладные расходы.
3. Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования	- Определение потенциального эффекта исследования

Перечень графического материала (с точным указанием обязательных чертежей):

1. Оценка конкурентоспособности технических решений
2. Матрица SWOT
3. Альтернативы проведения НИИ
4. График проведения и бюджет НИИ
5. Оценка ресурсной, финансовой и экономической эффективности НИИ

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

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН	Подопригора И.В.	к.э.н.		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8И6А	Степаненко В.В.		

ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа	ФИО
8И6А	Степаненко Владимиру Владимировичу

Школа	ИШИТР	Отделение (НОЦ)	ОИТ
Уровень образования	Бакалавриат	Направление/специальность	09.03.02 Информационные системы и технологии

Тема ВКР:

Серверное программное обеспечение информационной системы с технологией интерактивной визуализации средствами дополненной реальности	
Исходные данные к разделу «Социальная ответственность»:	
1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	<i>Разработка серверного программного обеспечения и клиентского веб-приложения для информационной системы с технологией интерактивной визуализации средствами дополненной реальности</i>
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
1. Правовые и организационные вопросы обеспечения безопасности: <ul style="list-style-type: none"> – специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства; – организационные мероприятия при компоновке рабочей зоны. 	Описание специальных правовых норм трудового законодательства и организационных мероприятий при работе с ПЭВМ
2. Производственная безопасность: 2.1. Анализ выявленных вредных и опасных факторов 2.2. Обоснование мероприятий по снижению воздействия	Вредные и опасные факторы: 1. отклонение показателей микроклимата в помещении 2. недостаточная освещенность рабочей зоны 3. умственное перенапряжение 4. опасность поражения электрическим током
3. Экологическая безопасность:	Влияние объекта исследования на экологию, мероприятия по защите окружающей среды
4. Безопасность в чрезвычайных ситуациях:	Возможные ЧС при разработке и эксплуатации исследуемого объекта, рассмотрение наиболее типичной ЧС

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

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент ООД	Немцова О.А.			

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8И6А	Степаненко В.В.		

РЕФЕРАТ

Выпускная квалификационная работа содержит 93 страницы, 23 рисунка, 21 таблиц, 15 источников.

Ключевые слова: сервер, веб-приложение, JavaScript, NodeJS, React, Express, информационная система.

Объектом исследования являются архитектура сервера и веб-приложения для информационной системы.

Целью данной работы является разработка серверного программного обеспечения и клиентского веб-приложения для информационной системы с технологией интерактивной визуализации средствами дополненной реальности.

В процессе исследования был проведен обзор существующих технологий для реализации серверной и клиентской частей веб-приложения, выбран стек технологий, разработана серверная и клиентская часть веб-приложения.

Разработанное веб-приложение позволяет клиентам создавать исходники (пары фото-видео), при помощи которых может быть осуществлен эффект «оживающих» фотографий.

Разработанный серверный компонент предоставляет доступ к исходникам мобильному приложению, реализующему технологию дополненной реальности.

Область применения: изготовление памятной фотопродукции.

В рамках развития проекта в будущем планируется следующее:

- оптимизация программного кода;
- расширение функциональных возможностей.

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

AR – augmented reality, дополненная реальность.

ИС – информационная система.

БД – база данных.

СУБД – система управления базами данных.

REST – Representational State Transfer, передача состояния представления.

API – Application Programming Interface, программный интерфейс приложения, интерфейс прикладного программирования.

ПО – программное обеспечение.

JWT – JSON Web Token.

URL – Uniform Resource Locator, унифицированный указатель ресурса.

SPA – Single Page Application, одностраничное приложение.

SEO – Search Engine Optimization, поисковая оптимизация.

UI – User Interface, интерфейс пользователя.

HTTP – HyperText Transfer Protocol, протокол передачи гипертекста.

JSON – JavaScript Object Notation.

XML – eXtensible Markup Language.

HTML – HyperText Markup Language, язык гипертекстовой разметки.

CSS – Cascading Style Sheets, каскадные таблицы стилей.

SQL – Structured Query Language, язык структурированных запросов.

Исходник – под исходником в данной работе была обозначена пара – фото и соответствующее ему видео, необходимые для создания эффекта «оживающей» фотографии.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	14
1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ.....	16
1.1 Описание информационной системы	16
1.2 Архитектура системы	17
1.3 Общие требования к функционалу системы	19
2. РАЗРАБОТКА СЕРВЕРНОГО КОМПОНЕНТА ИС	20
2.1 Определение требований к серверному компоненту ИС	20
2.2 Выбор СУБД.....	22
2.3 Проектирование БД	24
2.4 Выбор технологии для реализации серверного компонента ИС	26
2.4.1 NodeJS	26
2.4.2 PHP	26
2.4.3 ASP.NET.....	27
2.4.4 Python	27
2.4.5 Сравнительный анализ серверных технологий	28
2.5 Реализация серверного компонента ИС	30
2.5.1 Фреймворк Express.js.....	30
2.5.2 Аутентификация и авторизация	31
2.5.3 Хранение файлов.....	33
2.5.4 Взаимодействие с БД.....	34
2.5.5 Маршруты.....	36
3. РАЗРАБОТКА ВЕБ-КЛИЕНТА ИС	39
3.1 Определение требований к веб-клиенту.....	39

3.2 Выбор технологии для реализации веб-клиента	42
3.3 Проектирование пользовательского интерфейса	46
3.4 Разработка веб-клиента	50
3.4.1 Приложение на React.....	50
3.4.2 JSX	50
3.4.3 Компоненты.....	50
3.4.4 Взаимодействие с сервером.....	51
3.4.5 Авторизация	52
3.4.6 Маршрутизация.....	52
3.4.7 Реализация страниц приложения	54
4 ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ.....	61
4.1 Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения	61
4.1.1 Потенциальные потребители результатов исследования	61
4.1.2 Анализ конкурентных технических решений.....	61
4.1.3 Технология QuaD	63
4.1.4 SWOT-анализ.....	64
4.2 Планирование научно-исследовательских работ	65
4.2.1 Структура работ в рамках научного исследования	65
4.2.2 Определение трудоемкости выполнения работ	66
4.2.3 Разработка графика проведения научного исследования.....	67
4.2.4 Бюджет научно-технического исследования	69

4.2.4.1 Расчет материальных затрат научно-технического исследования.....	69
4.2.4.2 Расчет затрат на специальное оборудование для научных (экспериментальных) целей	70
4.2.4.3 Основная заработная плата исполнителей темы	71
4.2.4.4 Дополнительная заработная плата исполнителей темы .	73
4.2.4.5 Отчисления во внебюджетные фонды (страховые отчисления)	74
4.2.4.6 Накладные расходы	75
4.2.4.7 Формирование бюджета затрат научно-исследовательского проекта.....	75
4.3 Определение потенциального эффекта исследования.....	76
5 СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ.....	77
5.1 Введение	77
5.2 Правовые и организационные вопросы обеспечения безопасности	78
5.3 Производственная безопасность	81
5.4 Анализ выявленных вредных и опасных факторов.....	82
5.4.1 Отклонение показателей микроклимата в помещении	82
5.4.2 Недостаточная освещенность рабочей зоны	84
5.4.3 Умственное перенапряжение.....	85
5.4.4 Опасность поражения электрическим током	85
5.5 Экологическая безопасность	86
5.5.1 Влияние объекта исследования на окружающую среду	86
5.5.2 Мероприятия по обеспечению экологической безопасности	87

5.6 Безопасность в чрезвычайных ситуациях	87
5.7 Выводы по разделу	89
ЗАКЛЮЧЕНИЕ	90
CONCLUSION	91
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	92

ВВЕДЕНИЕ

Потребность в сохранении и передаче визуальной информации существовала всегда, и в течение многих веков только художники могли перенести реальность на холсты. Но с изобретением фотографии, скорость получения готового снимка и относительно низкая стоимость (в отличие от картины), дали возможность гораздо большему количеству людей сохранять важные события своей жизни в визуальной форме. Таким образом фотография вошла в повседневную жизнь людей.

Во времена, когда еще не были изобретены и распространены цифровые камеры, люди использовали пленочные фотоаппараты. После того, как были засняты кадры на такой фотоаппарат, чтобы получить фотографию нужно было проявить пленку. В результате проявление пленки, на выходе получалась фотография, в виде объекта реального мира, который можно подержать в руках.

С приходом цифровой революции, начавшейся в 1980-х годах, фотокамеры научились создавать цифровые изображения. Все современные смартфоны оснащены фото-видео камерами, при помощи которых ежедневно делаются миллионы фотографий. Люди получили возможность мгновенно создавать фотографии, просматривать и делиться ими, без необходимости предварительной печати.

Таким образом начались два параллельных процесса:

- существенно уменьшается количество печатаемых фотографий (зачем печатать, если можно смотреть на компьютере и передавать по интернету);
- существенно увеличивается количество создаваемых фотографий в единицу времени.

Но все же, по-прежнему печатные фотографии остаются актуальными, например, при создании памятных фотоальбомов.

В рамках данной работы было выполнено исследование, проектирование и разработка системы, обеспечивающей эффект дополненной реальности на изображении с камеры мобильного устройства. Информационная система позволяет пользователю «оживлять» фотографии, при помощи камеры мобильного телефона, средствами дополненной реальности, заменяя изображения фотографии на экране телефона соответствующим видео.

Таким образом, объем информации, передаваемый обычной фотографией или стендом может быть увеличен за счет добавляемых объектов – текста, изображений, видеороликов, аудиозаписей. При помощи этого может быть снова актуализирована печать фотографий. Кроме того, данная система может быть использована для повышения эффективности печатной рекламной продукции, позволяя увеличить ее информативность.

Целью данной работы является разработка серверного программного обеспечения и клиентского веб-приложения для информационной системы с технологией интерактивной визуализации средствами дополненной реальности. Разработанные компоненты позволяют пользователям создавать исходники (пары фото-видео) и получать их для «оживления» фотографий при помощи мобильного приложения.

В соответствии с поставленной целью, требовалось решить следующие задачи:

- определить требования к серверному компоненту;
- спроектировать БД;
- разработать серверный компонент;
- определить требования к веб-клиенту;
- спроектировать пользовательский интерфейс;
- разработать клиентское веб-приложение.

1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

1.1 Описание информационной системы

Дополненная реальность (augmented reality, AR) – результат введения в поле восприятия любых сенсорных данных с целью дополнения сведений об окружении и улучшения восприятия информации.

Разрабатываемая информационная система позволит, при помощи смартфона, благодаря использованию дополненной реальности, «оживлять» фотографии.

Под «оживлением» фотографии понимается замена изображения фотографии на экране телефона, полученного при помощи камеры, соответствующей видеозаписью. Таким образом, подмена фотографии на видеозапись создает впечатление того, что фотография оживает.

В результате упрощенный алгоритм оживления фотографии выглядит следующим образом:

- пользователь наводит камеру смартфона на фотографию;
- система распознает фотографию;
- система находит соответствующую фотографии видеозапись;
- система заменяет изображение фотографии на видеозапись.

В данной работе рассматривается проектирование базы данных, разработка серверной части и веб-клиента описанной информационной системы.

1.2 Архитектура системы

Помимо непосредственно мобильного приложения, которое создает дополненную реальность (разработка мобильного приложения не рассматривается в данной работе) в информационную систему входят:

1. база данных;
2. серверное приложение (веб-сервер);
3. веб-клиент.

В базе данных хранится информация о пользователях и информация необходимая для передачи изображений и видеозаписей на клиентские приложения.

Серверное приложение предоставляет функционал, позволяющий клиентским приложениям загружать и скачивать фотографии и видеозаписи (так как видеозаписи обычно имеют большой объем, принято решение хранить их на сервере), регистрироваться и авторизовываться в системе.

Веб-клиент предоставляет удобные интерфейсы для регистрации, авторизации и создания пар фото-видео, являющихся исходными данными для реализации дополненной реальности.

Общая архитектура информационной системы представлена на рисунке 1.

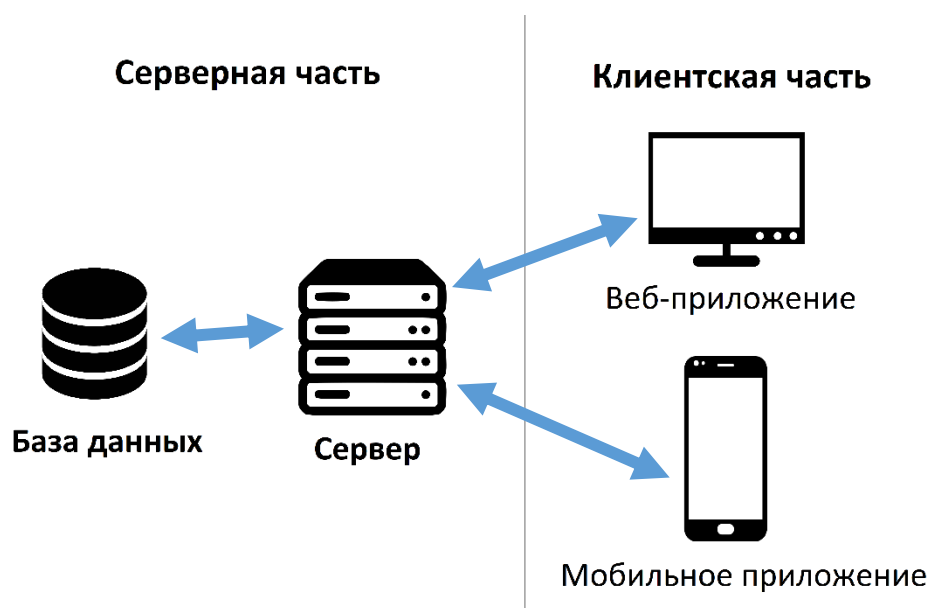


Рисунок 1 – Общая архитектура ИС

Для передачи клиентам данных из базы данных и файлов из файловой системы хранилища, веб-сервер обеспечивает поддержку HTTP запросов.

В качестве формата данных для обмена между клиентскими приложениями и сервером, выбран JSON, по следующим причинам:

- компактность;
- легкая читаемость;
- легкость в преобразовании к структуре данных;
- наличие во многих языках функций и библиотек для чтения и создания структур JSON.

1.3 Общие требования к функционалу системы

В разрабатываемой информационной системе присутствует всего три типа пользователей: авторизованный пользователь мобильного приложения, авторизованный пользователь веб-приложения, неавторизованный пользователь.

Неавторизованный пользователь имеет следующие функциональные возможности:

- зарегистрироваться в системе через веб-приложение;
- зарегистрироваться в системе через мобильное приложение;
- авторизоваться в системе через веб-приложение;
- авторизоваться в системе через мобильное приложение.

Авторизованный пользователь веб-приложения имеет следующие функциональные возможности:

- создание исходника для «живой» фотографии;
- создание альбома «живых» фотографий;
- удаление фотографии;
- удаление альбома;
- просмотр своих исходников для оживления фотографий;
- просмотр списка альбомов;
- получение идентификатора альбома (для предоставления возможности поделиться альбомом);
- добавление альбома по идентификатору;
- выход из учетной записи.

Авторизованный пользователь мобильного приложения имеет следующие функциональные возможности:

- «оживление» фотографии, при помощи мобильного приложения;
- выход из учетной записи.

2. РАЗРАБОТКА СЕРВЕРНОГО КОМПОНЕНТА ИС

2.1 Определение требований к серверному компоненту ИС

В информационной системе присутствуют такие компоненты как мобильное приложение и веб-клиент. Для того, чтобы эти компоненты могли получать данные из базы данных и файловой системы хранилища, реализован серверный компонент, который передает данные клиентским приложениям. Серверный компонент реализует REST API.

REST – это общие принципы организации взаимодействия клиентского приложения с сервером посредством протокола HTTP. Особенность REST в том, что сервер не запоминает состояние пользователя между запросами – в каждом запросе передаётся информация, идентифицирующая пользователя (например, token, полученный через OAuth-авторизацию) и все параметры, необходимые для выполнения операции.

Всё взаимодействие с сервером сводится к 4 операциям (4 – это необходимый и достаточный минимум, в конкретной реализации типов операций может быть больше):

- получение данных с сервера (обычно в формате JSON, или XML);
- добавление новых данных на сервер;
- модификация существующих данных на сервере;
- удаление данных на сервере.

В соответствии с архитектурой и общим требованиям к информационной системе были составлены требования к серверному компоненту, требования представлены в таблице 1.

Таблица 1 – Требования к серверному компоненту ИС

Требование	Комментарий
Поддержка протокола HTTP	Протокол HTTP используется в качестве протокола обмена информацией между сервером и клиентскими приложениями

Реализация REST API	REST используется, так как в информационной системе присутствует мобильное приложение, которое отправляет и запрашивает данные. Кроме того, клиентскому веб-приложению от сервера нужны только данные, отрисовка HTML происходит на клиенте.
JSON в качестве формата данных	В качестве формата данных выбран JSON, так как он удобен, кроме того, является нативным форматом данных для JavaScript.
Поддержка multipart/form-data	Используется для передачи файлов (изображений и видео) с клиента.
Хранение изображений и видео в файловой системе	Было принято решение хранить файлы в файловой системе сервера, а не в БД, так как таким образом к ним обеспечивается более быстрый доступ.

2.2 Выбор СУБД

В разрабатываемой информационной системе предполагается хранить в базе данных информацию о пользователях системы и информацию о их фото и видео, позволяющую определить пользователя, которому они принадлежат и получить их с сервера.

Важным этапом при проектировании информационной системы является выбор СУБД. Совокупность программных и лингвистических средств общего или специального назначения, обеспечивающих управление созданием и использованием баз данных [1].

Система обеспечивает безопасность, надёжность хранения и целостность данных, а также предоставляет средства для администрирования БД.

СУБД, базирующиеся на использовании реляционной модели данных, называют реляционными СУБД.

Реляционные СУБД (SQL) хранят структурированные данные, которые обычно представляют объекты реального мира. Это могут быть сведения о человеке, или о содержимом корзины для товаров в магазине, сгруппированные в таблицах, формат которых задан на этапе проектирования хранилища.

Нереляционные СУБД (NoSQL) устроены иначе. Например, документо-ориентированные базы хранят информацию в виде иерархических структур данных. Речь может идти об объектах с произвольным набором атрибутов. То, что в реляционной БД будет разбито на несколько взаимосвязанных таблиц, в нереляционной может храниться в виде целостной сущности [2].

Также при сравнении СУБД можно разделить их на коммерческие (требуется лицензия) и свободные СУБД (лицензия не требуется).

Среди наиболее популярных СУБД в настоящее время можно выделить и рассмотреть следующие: MS SQL, MySQL, PostgreSQL, MongoDB. Сравнение данных СУБД приведено в таблице 2.

Таблица 2 – Сравнение СУБД

СУБД	Схема данных	Бесплатное использование	Библиотека объектного моделирования
MS SQL	Реляционная	Нет	Sequelize
MySQL	Реляционная	Да	Sequelize
PostgreSQL	Реляционная	Да	Sequelize
MongoDB	NoSQL	Да	Mongoose

MongoDB является документоориентированной СУБД, с открытым исходным кодом. NoSQL СУБД проявляют свои положительные стороны в основном при работе с большими объемами неструктурированной информации.

Так как в данном проекте база данных будет небольшой, был сделан выбор в пользу реляционных СУБД, предоставляющих большую безопасность и структурированность данных.

Из трех рассматриваемых реляционных СУБД, бесплатными к использованию являются MySQL и PostgreSQL, для использования MS SQL необходимо приобрести лицензию.

Из двух оставшихся вариантов, был выбран PostgreSQL, по причине того, что PostgreSQL инструмент более мощный, более гибкий и имеет хорошую документацию.

2.3 Проектирование БД

После определения основных требований к системе и выбора СУБД, необходимо определить какие данные будут храниться в БД и разработать схему БД.

При рассмотрении требований системы можно выделить следующие основные сущности:

- пользователь;
- пара фотография-видео;
- альбом.

Основными атрибутами сущности «Пользователь» являются:

1. идентификатор пользователя;
2. email;
3. хеш пароля.

Основными атрибутами сущности «Пара» являются:

1. идентификатор изображения;
2. идентификатор видео;
3. название пары;
4. дата создания.

Основными атрибутами сущности «Альбом» являются:

1. идентификатор альбома;
2. название альбома;
3. дата создания.

Концептуальная модель базы данных представлена на рисунке 2.

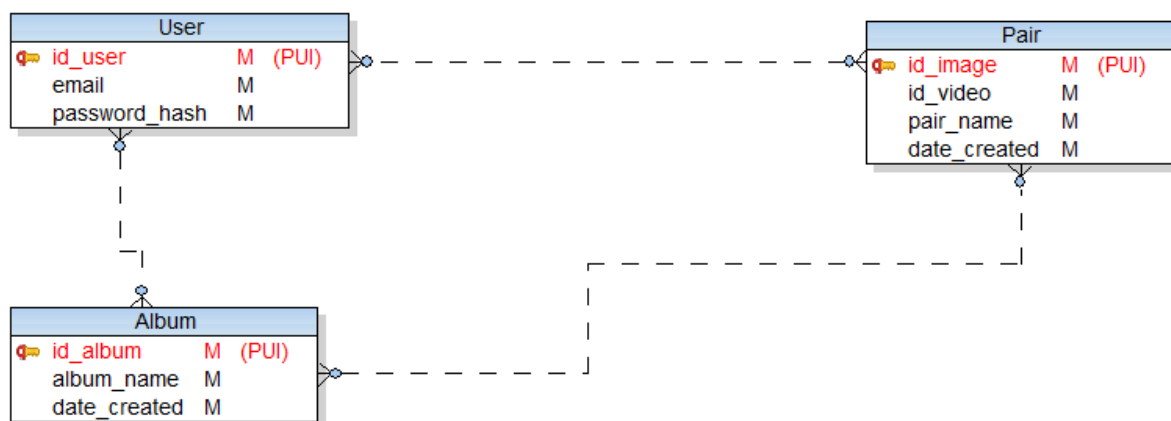


Рисунок 2 – Концептуальная модель БД

На рисунке 3 представлена физическая модель БД для СУБД PostgreSQL.

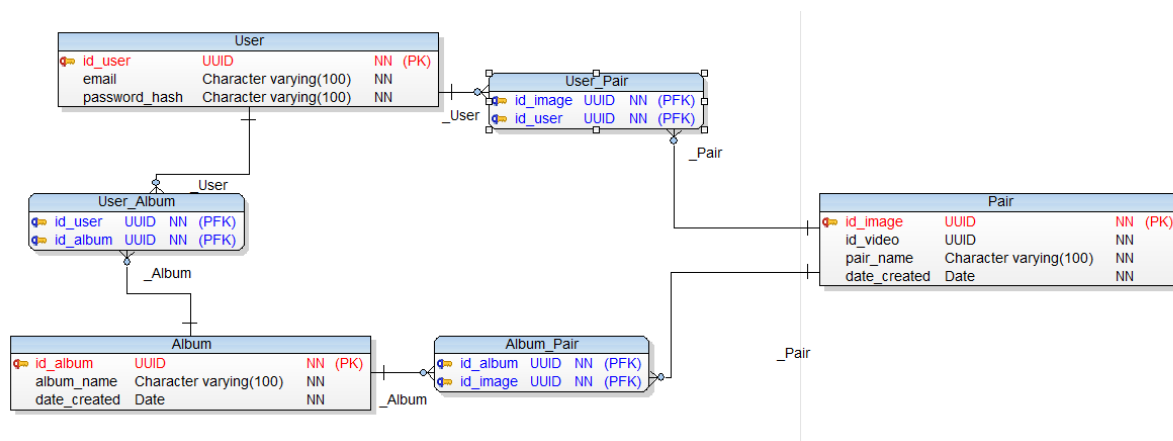


Рисунок 3 – Физическая модель БД

2.4 Выбор технологии для реализации серверного компонента ИС

2.4.1 NodeJS

NodeJS – это среда, позволяющая запускать код на Javascript на серверной стороне. Разрабатываемые скрипты могут быть запущены на машинах под управлением как Windows, так и Linux.

Несмотря на то, что чаще всего такие решения используются для разработки веб-приложений, десктопное ПО также может быть реализовано на NodeJS.

В качестве преимуществ можно выделить следующие характеристики платформы:

- асинхронность (несмотря на то, что NodeJS является однопоточным языком программирования, механизмы функций обратного вызова, Promise-конструкций и обработка событий позволяют разработчику писать асинхронный код);
- высокая масштабируемость (поскольку он использует механизм событий, Node.js обладает высокой масштабируемостью и помогает серверу в неблокирующем ответе);
- высокая скорость выполнения кода (поскольку Node.js построен на движке JavaScript Google Chrome V8, его библиотеки очень продвинуты и, следовательно, способны выполнять код с большей скоростью);
- open-source (NodeJS распространяется под лицензией MIT X11. Таким образом, открытый исходный код платформы может быть изучен, отредактирован и распространен для любых целей).

2.4.2 PHP

PHP – один из первых языков, на которых разрабатывались веб-приложения. Этот язык программирования изначально ориентирован на генерацию HTML-страниц и формирование ответов на HTTP-запросы.

PHP имеет внушительный объем документации и, благодаря внушительной истории, обширное сообщество разработчиков.

Значимое преимущество PHP — это его гибкость и универсальность. Он совместим со всеми основными платформами: от Windows, Unix и Linux до MacOS; поддерживает большинство серверов (включая Apache, Microsoft IIS, Netscape, iPlanet, Caudium, Xitami и Tornado) и более 20 баз данных (среди которых — MySQL, MongoDB, PostgreSQL и др.)

2.4.3 ASP.NET

ASP.NET (Active Server Pages для .NET) – платформа разработки веб-приложений, в состав которой входит: веб-сервисы, программная инфраструктура, модель программирования, от компании Майкрософт [3].

Благодаря плотной интеграции со средой разработки и возможностью написания кода веб-приложений на всех языках платформы “.NET”, именно ASP.NET является наиболее подходящим решением для разработки сложных систем со сложной логикой и множеством отдельных модулей с возможностью запуска на разных платформах.

Фреймворк .NET также является программным обеспечением с открытым исходным кодом [4]. Модули .Net Core поддерживаются на репозиториях хостинга Github, и в основном используют лицензии MIT и Apache 2.

2.4.4 Python

Изначально Python был разработан как язык, ориентированный на повышение производительности разработчика и читаемости кода.

Python имеет упрощенный минималистичный синтаксис и отлично подходит для изучения новичками. В тоже время существует большое количество подключаемых библиотек, что позволяет разрабатывать приложения с самым широким функционалом.

Программирование на Python подразумевает обязательное использование иерархических отступов в коде. Этот язык аргументированно является самым читабельным.

В следствие того, что концепция высокоуровневого языка делает упор на скорость разработки, скорость выполнения программ на Python зачастую меньше, чем на компилируемых языках.

2.4.5 Сравнительный анализ серверных технологий

В таблице 3 представлено сравнение рассмотренных платформ для разработки серверных веб-приложений по наличию отдельных характеристик.

Таблица 3 – Сравнение платформ разработки сервера приложений

	Python	NodeJS	PHP	ASP.net
Строгая типизация	—	—	—	+
Кроссплатформенность	+	+	+	+
Широкая направленность	+	+	—	+
Объектно-ориентированное программирование	+	+	+	+
Событийно-ориентированное программирование	+	+	—	+
Открытый исходный код	—	+	—	+
Высокий уровень документированности	—	+	+	—

Анализ преимуществ и недостатков рассмотренных вариантов показал, что язык программирования PHP является узкопрофильным, а значит менее перспективным. Несмотря на ряд преимуществ языка Python, таких как упрощенный синтаксис и возможность подключения библиотек на языке C, сравнительно низкая скорость выполнения является существенным недостатком. Перспектива широкого распространения платформы NodeJS

делает её более предпочтительной для изучения и последующей разработки [5].

В данной информационной системе было принято решение разрабатывать серверный компонент на платформе NodeJS. Важным фактором при осуществлении выбора является то, что клиентская и серверная часть веб-приложения разрабатываются одним разработчиком, использования JavaScript в обеих частях приложения значительно упростит задачу.

2.5 Реализация серверного компонента ИС

2.5.1 Фреймворк Express.js

Используя Express и Node.js, можно реализовать полноценный API в стиле REST для взаимодействия с пользователем. Архитектура REST предполагает применение следующих методов или типов запросов HTTP для взаимодействия с сервером:

- GET;
- POST;
- PUT;
- DELETE.

Зачастую REST-стиль особенно удобен при создании всякого рода Single Page Application.

Express.js, или просто Express, фреймворк web-приложений для Node.js, реализованный как свободное и открытое программное обеспечение под лицензией MIT. Он спроектирован для создания веб-приложений и API. Де-факто является стандартным каркасом для Node.js.

Express — это минималистичный и гибкий веб-фреймворк для приложений Node.js, предоставляющий обширный набор функций для веб-приложений. Имея в своем распоряжении множество служебных методов HTTP и промежуточных обработчиков, создать надежный API можно быстро и легко.

Express — это веб-фреймворк маршрутизации и промежуточной обработки с минимальной собственной функциональностью: приложение Express, по сути, представляет собой серию вызовов функций промежуточной обработки.

Функции промежуточной обработки (middleware) — это функции, имеющие доступ к объекту запроса (req), объекту ответа (res) и к следующей функции промежуточной обработки в цикле “запрос-ответ” приложения.

Следующая функция промежуточной обработки, как правило, обозначается переменной `next`.

Функции промежуточной обработки могут выполнять следующие задачи:

- выполнение любого кода;
- внесение изменений в объекты запросов и ответов;
- завершение цикла «запрос-ответ»;
- вызов следующей функции промежуточной обработки из стека.

Если текущая функция промежуточной обработки не завершает цикл «запрос-ответ», она должна вызвать `next()` для передачи управления следующей функции промежуточной обработки. В противном случае запрос зависнет.

Можно использовать как готовые модули, так и создавать свои собственные `middleware` и встраивать их в стек функций.

Например, для того чтобы облегчить работу с запросами по методу `POST` к `API`, было использовано промежуточное ПО для разбора тела запроса, готовый модуль `body-parser`. За счет этого модуля будет извлечено все тело пришедшего запроса, а его данные преобразованы в `JSON`-объект, с которым мы можем работать.

Для защиты некоторых маршрутов, от использования неавторизованными пользователями был реализован собственный модуль `check-auth`.

2.5.2 Аутентификация и авторизация

Так как в разрабатываемой информационной системе имеется разграничение пользователей по личным учетным записям, необходимо реализовать процесс аутентификации и авторизации.

Аутентификация – процедура проверки подлинности.

Авторизация – процедура предоставления субъекту определённых прав.

Так как REST API не сохраняют информацию о состоянии, это означает, что не может быть организована сессия. Следовательно, раз состояние аутентификации пользователя не может быть сохранено в сессиях, каждый запрос должен приходить вместе с определенным видом параметров аутентификации. Общепринятая практика состоит в том, что для аутентификации пользователя с каждым запросом отправляется токен доступа.

Токен представляет собой зашифрованную строку. Использование токена обосновывается тем, что кража токена влечет за собой только кражу временного доступа к серверу, в то время как кража пароля означает полный доступ к учетной записи.

Одним из стандартов создания токенов является JSON Web Token. JSON Web Token (JWT) – это открытый стандарт для создания токенов доступа, основанный на формате JSON. Как правило, используется для передачи данных для аутентификации в клиент-серверных приложениях. Токены создаются сервером, подписываются секретным ключом и передаются клиенту, который в дальнейшем использует данный токен для подтверждения своей личности. Секретный ключ хранится только на сервере и позволяет зашифровать и расшифровать токен.

Токен JWT состоит из трех частей: заголовка (header), полезной нагрузки (payload) и подписи или данных шифрования. В заголовке указывается необходимая информация для описания самого токена. В секции полезная нагрузка указывается пользовательская информация (например, имя пользователя и уровень его доступа), а также могут быть использованы некоторые служебные ключи, все они являются необязательными. Так же в полезной нагрузке может передаваться время, по истечению которого токен станет не валидным.

При разработке сервера был использован готовый пакет jsonwebtoken, который значительно упрощает реализацию стандарта JWT.

Была реализована функция промежуточной обработки check-auth, которая проверяет наличие в заголовке запроса токена, в случае его наличия валидирует токен и в случае успеха передает запрос в следующую функцию, иначе отправляет клиенту ошибку 401 Unauthorized. Данный модуль был встроен во все необходимые маршруты.

Для безопасного хранения паролей клиентов, в базе хранится только хеш пароля. Для получения хеша использована библиотека bcrypt.

2.5.3 Хранение файлов

В разрабатываемой информационной системе необходимо хранить большое количество файлов от пользователей (фото и видео). Было принято решение хранить файлы в файловой системе сервера, а не в БД, так как это позволяет получать к ним прямой доступ, и повышает скорость работы приложения.

В БД хранятся только идентификаторы файлов, для того чтобы построить к ним путь в случае запроса от пользователя.

Структура директорий для хранения файлов представлена на рисунке 4.

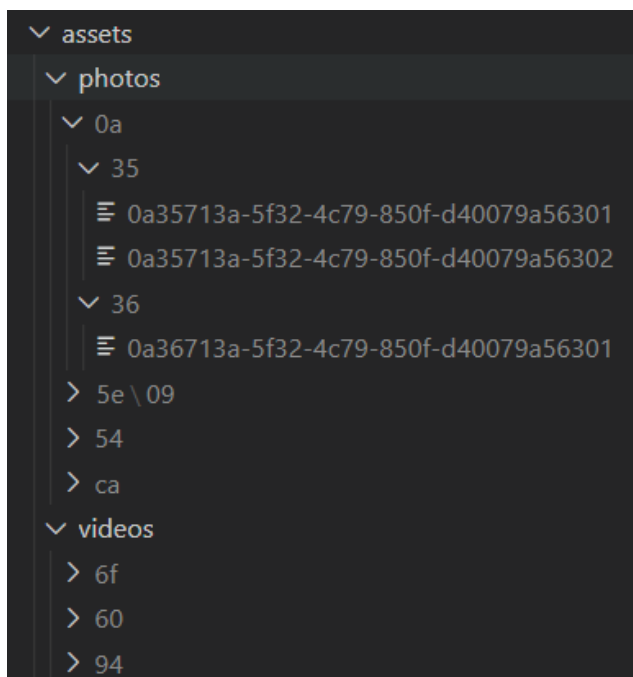


Рисунок 4 – Структура директорий для хранения файлов

Известно, что в случае, когда в одной папке хранится много файлов и папок, то доступ к ним замедляется (проблемы могут возникать уже от 1000 файлов и папок). Для решения этой проблемы, файлы хранятся в такой иерархии, чтобы не происходило их накопление в одной папке.

Внутри папки `assets` находятся папки `photos` и `videos`, внутри которых создаются папки для файлов в два уровня. Например, для фотографии, внутри папки `photos` будет создана папка, название которой будет состоять из первых двух символов названия фотографии. Внутри этой папки будет создана вторая папка, название которой будет состоять из третьего и четвертого символов названия фотографии. Далее будет находиться сам файл. То есть для файла фотографии с названием «0a35713a-5f32-4c79-850f-d40079a56301» путь будет следующим: `assets/photos/0a/35/0a35713a-5f32-4c79-850f-d40079a56301`.

Так как в качестве названия файла используется UUID, который представлен в виде шестнадцатеричных символов, один символ может иметь 16 разных значений. То есть в папке `photos` будет не более 256 других папок, по причине того, что в их названии используется два символа. В папках, находящихся на уровень глубже, также будет не более 256 других папок.

Таким образом, благодаря уровню вложенности в 2 папки, для файлов может быть $256 * 256 = 65536$ папок. Тогда для одного миллиона файлов, в каждой папке второго уровня вложенности будет в среднем около 15 файлов.

Такая структура хранения файлов позволяет получать к ним быстрый доступ даже при большом их количестве.

2.5.4 Взаимодействие с БД

В качестве СУБД для разрабатываемой ИС была выбрана PostgreSQL.

Для взаимодействия с БД была использована библиотека `node-postgres`. `Node-postgres` – это неблокирующий PostgreSQL клиент для Node.js.

Большой ошибкой с точки зрения безопасности является ввод данных, пришедших от пользователя, в том виде, в котором они были переданы, так как БД станет уязвимой к SQL инъекциям.

SQL инъекция – один из распространённых способов взлома сайтов и программ, работающих с базами данных, основанный на внедрении в запрос произвольного SQL-кода.

Внедрение SQL, в зависимости от типа используемой СУБД и условий внедрения, может дать возможность атакующему выполнить произвольный запрос к базе данных (например, прочитать содержимое любых таблиц, удалить, изменить или добавить данные), получить возможность чтения и/или записи локальных файлов и выполнения произвольных команд на атакуемом сервере.

Атака типа внедрения SQL может быть возможна из-за некорректной обработки входных данных, используемых в SQL-запросах.

Функция `pool.query` из библиотеки `node-postgres`, которая позволяет осуществлять запросы к БД имеет встроенную защиту от SQL инъекций.

С использованием библиотеки `node-postgres`, были созданы следующие модели для взаимодействия с БД:

- `user`;
- `album`;
- `pair`.

Модель `user` содержит следующие функции:

- `createUser` (позволяет создать нового пользователя);
- `findByEmail` (позволяет найти пользователя по email).

Модель `pair` содержит следующие функции:

- `getUserPairs` (позволяет получить информацию об источниках пользователя: идентификатор картинки, идентификатор видео, название источника, дата создания);
- `createPair` (позволяет создать новый источник);

- `addUserPair` (позволяет добавить существующий источник пользователю);
- `deleteUserPair` (позволяет удалить источник из списка источников пользователя).

Модель `album` содержит следующие функции:

- `createAlbum` (позволяет создать новый альбом);
- `getUserAlbums` (позволяет получить информацию обо всех альбомах пользователя: идентификатор альбома, название альбома, дата создания);
- `deleteUserAlbum` (позволяет удалить альбом из списка альбомов пользователя);
- `addUserAlbum` (позволяет добавить альбом в список альбомов пользователя);
- `addAlbumPair` (позволяет добавить источник в альбом).

2.5.5 Маршруты

Для предоставления клиентам необходимого функционала, на сервере были реализованы все требуемые маршруты. Информация о созданных маршрутах приведена в таблице 4.

Таблица 4 – Маршруты, реализованные на сервере.

Маршрут	Метод	Проверка токена	Описание
/user/signup	POST	Нет	Данный маршрут позволяет клиенту создать новую учетную запись, в случае успеха возвращается ответ с подтверждением, иначе возвращается ответ с ошибкой.

/user/signin	POST	Нет	Данный маршрут позволяет клиенту получить токен авторизации. Для этого клиент передает email и пароль, в ответ получает токен.
/user/profile	GET	Да	Данный маршрут позволяет произвести валидацию токена.
/img	GET	Да	Данный маршрут позволяет клиенту получить информацию о всех добавленных в его учетной записи исходниках, личных и из альбомов: идентификатор картинки, идентификатор видео, название исходника, дату создания.
/img/own	GET	Да	Данный маршрут позволяет клиенту получить информацию о своих личных исходниках.
/img/:id	GET	Нет	Данный маршрут позволяет клиенту получить изображение по его идентификатору.
/img	POST	Да	Данный маршрут позволяет клиенту создать новый исходник. Для этого клиент передает название исходника, картинку и видеозапись.
/img	DELETE	Да	Данный маршрут позволяет клиенту удалить исходник из списка исходников своей учетной записи.

/album	POST	Да	Данный маршрут позволяет клиенту создать новый альбом. Для этого клиент передает массив фотографий, массив видео, массив названий исходников и название альбома.
/album/:id	POST	Да	Данный маршрут позволяет клиенту добавить в список своих альбомов альбом по идентификатору.
/album	GET	Да	Данный маршрут позволяет клиенту получить информацию о всех добавленных в его учетной записи альбомах: идентификатор альбома, название альбома, дата создания.
/album	DELETE	Да	Данный маршрут позволяет клиенту удалить альбом из списка альбомов своей учетной записи.
/video/stream/:id	GET	Нет	Данный маршрут позволяет клиенту получить видеозапись по ее идентификатору.

3. РАЗРАБОТКА ВЕБ-КЛИЕНТА ИС

3.1 Определение требований к веб-клиенту

Одним из компонентов разрабатываемой информационной системы является веб-клиент, позволяющий взаимодействовать с веб-сервером при помощи браузера.

Главной задачей клиентского веб-приложения является предоставление пользователю удобного интерфейса для создания исходников (пар фото-видео), на основе которых при помощи мобильного приложения будет реализовываться эффект «оживающих» фотографий.

В соответствии с общими требованиями к информационной системе были разработаны требования к клиентскому приложению, требования представлены в таблице 5.

Таблица 5 – Требования к веб-приложению

Требование	Комментарий
Возможность регистрации	Интерфейс регистрации реализуется в виде отдельной страницы с формой, в которой присутствуют поля «электронная почта» и «пароль». После успешной регистрации пользователь получает возможность авторизоваться в системе под своей учетной записью.
Возможность авторизации	Интерфейс авторизации реализуется в виде отдельной страницы с формой, в которой присутствуют поля «электронная почта» и «пароль». В случае успешной авторизации пользователю становится доступен функционал авторизованного пользователя, иначе выводится сообщение об ошибке.
Возможность выхода из учетной записи	Реализуется в виде кнопки «Выйти» на каждой странице авторизованного пользователя. При нажатии на кнопку осуществляется выход из учетной записи.

Возможность добавить новый источник (фото- видео)	Реализуется в виде страницы на которой представлена форма, содержащая поля для ввода названия источника, добавления фотографии, добавления видео и кнопки сохранить.
Возможность просмотреть добавленные в систему источники (фото-видео)	Реализуется в виде страницы на которой предствлены карточки добавленных источников. В карточке отображаются фотография, названия источника, дата создания. При наведении курсора на карточку, на месте фотографии запускается видео из источника.
Возможность удалить источник	Реализуется в виде кнопки в карточке источника, при нажатии на кнопку источник удаляется.
Возможность создания альбома	Реализуется в виде страницы с формой, на которой представлено поле для ввода названия альбома и кнопки «Сохранить». Отдельным блоком отображаются поля для ввода названия источника, добавления фотографии и видео и кнопки «Добавить». Этот блок позволяет добавить несколько источников, при нажатии кнопки «Добавить», добавляется и отображается на странице в виде карточки новый источник. Нажатие кнопки «Сохранить» позволяет создать альбом со всеми добавленными источниками.
Возможность просмотреть добавленные альбомы	Реализуется в виде страницы на которой предствлены карточки добавленных альбомов. В карточке отображается назване альбома и дата создания.
Возможность поделиться альбомом	Реализуется посредством обмена между пользователями идентификаторами альбомов. При нажатии на кнопку копирования идентификатора,

	<p>расположенную в карточке товара, идентификатор копируется в буфер обмена пользователя. Данный идентификатор можно передать другому пользователю. Пользователь, получивший идентификатор, может вставить его в соответствующее поле на странице отображения альбомов и нажатием кнопки «Добавить» получить альбом.</p>
Возможность удалить альбом	<p>Реализуется в виде кнопки в карточке альбома, при нажатии на кнопку альбом удаляется.</p>
Адаптация под мобильные устройства	<p>Реализуется посредством использования библиотеки реализующей отзывчивый веб-дизайн.</p>

3.2 Выбор технологии для реализации веб-клиента

Для реализации клиентской части веб-приложений используются такие технологии как HTML, CSS и JavaScript.

HTML – это стандартный язык разметки, используемый для создания веб-страниц и веб-приложений. Его элементы определяют стандартные блоки страниц, а также представляют форматированный текст, изображения, формы ввода данных и другие структуры. Когда браузер направляет запрос URL-адреса для получения страницы или приложения, в первую очередь возвращается HTML-документ. Этот HTML-документ может содержать ссылку или дополнительные сведения об оформлении или макете в виде CSS, а также о поведении в виде JavaScript.

Каскадные таблицы стилей (CSS) определяют оформление и макет элементов HTML. Стили CSS могут применяться непосредственно к элементу HTML, отдельно определяться на той же странице или определяться в отдельном файле, ссылка на который будет задаваться на странице.

JavaScript – это динамически интерпретируемый язык программирования, который был стандартизирован в спецификации ECMAScript. Это язык программирования для Интернета. Как и CSS, JavaScript может определяться в качестве атрибутов в элементах HTML, в виде блоков скрипта на странице, а также в отдельных файлах [6].

В настоящее время существует два принципиальных подхода к созданию веб-приложений:

- многостраничные веб-приложения (традиционные);
- одностраничные приложения (single page application, SPA).

Многостраничные приложения имеют более классическую архитектуру. Каждая страница отправляет запрос на сервер и полностью обновляет все данные. Даже если эти данные небольшие. Таким образом тратится производительность на отображение одних и тех же элементов. Соответственно это влияет на скорость и производительность.

Преимущества многостраничных приложений:

- простая SEO оптимизация – можно оптимизировать каждую из страниц приложения под нужные ключевые запросы;
- работа в браузерах без поддержки JavaScript.

Одностраничное веб-приложение загружается на одну HTML-страницу. Благодаря динамическому обновлению с помощью JavaScript, во время использования не нужно перезагружать или подгружать дополнительные страницы. На практике это означает, что пользователь видит в браузере весь основной контент, а при прокрутке или переходах на другие страницы, вместо полной перезагрузки нужные элементы просто подгружаются.

В процессе работы пользователю может показаться, что он запустил не веб-сайт, а десктопное приложение, так как оно мгновенно реагирует на все его действия, без задержек и «подвисаний».

Преимущества одностраничных приложений:

- отличное быстродействие (так как большинство ресурсов, которые они используют (HTML+CSS+Скрипты), загружаются лишь однажды в течение сессии использования приложения);
- экономия трафика;
- возможность работы с одним бэкендом вместе с мобильным приложением;
- разработка проще и эффективнее (так как приложение может быть четко разделено на фронтенд и бэкенд);
- проще отлаживать.

В данной информационной системе было принято решение реализовать одностраничное веб-приложение, так как это позволит использовать один бэкенд вместе с мобильным приложением, кроме того, одностраничные приложения обладают большим быстродействием.

Для облегчения процесса написания и запуска веб-приложений используются веб-фреймворки и библиотеки. Фреймворки – это программные продукты, которые упрощают создание и поддержку технически сложных или нагруженных проектов. Фреймворк, как правило, содержит только базовые программные модули, а все специфичные для проекта компоненты реализуются разработчиком на их основе. Тем самым достигается не только высокая скорость разработки, но и большая производительность и надёжность решений.

Среди наиболее популярных в настоящее время выделяются веб-фреймворки Angular, Vue.js и библиотека React. Их сравнение приведено в таблице 6.

Таблица 6 – Сравнение фреймворков и библиотек для реализации клиентской части веб-приложения

Фреймворк/ библиотека	Плюсы	Минусы
React	Поддержка Facebook Virtual DOM Легкость в изучении JSX Большое сообщество разработчиков	Не фреймворк Разные подходы к разработке
Vue.js	Virtual DOM Маленький размер Хорошая документация	Маленькое сообщество разработчиков
Angular	Поддержка Google Строгая типизация Хорошая документация	Маленькое сообщество разработчиков Сложность в изучении Обязательное использование Typescript

В данной работе для разработки пользовательских интерфейсов была выбрана библиотека React, по причине того, что Angular больше подходит к большим проектам, а Vue имеет небольшое сообщество разработчиков.

React (иногда React.js или ReactJS) – JavaScript-библиотека с открытым исходным кодом для разработки пользовательских интерфейсов.

React разрабатывается и поддерживается Facebook, Instagram и сообществом отдельных разработчиков и корпораций.

React может использоваться для разработки одностраничных и мобильных приложений. Его цель – предоставить высокую скорость, простоту и масштабируемость. В качестве библиотеки для разработки пользовательских интерфейсов React часто используется с другими библиотеками, такими как Redux и GraphQL [7].

3.3 Проектирование пользовательского интерфейса

Интерфейс пользователя, он же пользовательский интерфейс (UI – англ. user interface) – интерфейс, обеспечивающий передачу информации между пользователем-человеком и программно-аппаратными компонентами компьютерной системы.

Правильно спроектированный пользовательский интерфейс не только решает нашу проблему взаимодействия с приложением, но и делает это взаимодействие максимально комфортным. Нам важно наличие интерфейса, позволяющего при меньшем количестве усилий ознакомиться с возможностями приложения и понять основные принципы работы в нём.

Ясность – это первая и самая главная задача любого интерфейса. Чтобы интерфейс эффективно помогал людям добиваться своих целей, он должен обладать следующими характеристиками:

- интерфейс должен быть узнаваемым, а его назначение – очевидным для пользователя;
- люди должны понимать, с чем они взаимодействуют через интерфейс;
- процесс взаимодействия с интерфейсом должен быть предсказуемым.

Ясность рождает в пользователях уверенность и готовность продолжать работу с интерфейсом. Сто понятных экранов лучше, чем один беспорядочный.

Ниже представлен разработанный для создаваемого веб-приложения макет пользовательского интерфейса.

После входа на сайт, пользователь попадает на страницу регистрации (рисунок 5). Здесь он может, заполнив поля и нажав кнопку зарегистрироваться, создать новый аккаунт, либо нажатием на ссылку «Войти в аккаунт» перейти на страницу авторизации (рисунок 6).

Регистрация

*e-mail

*Пароль

*Повторите пароль

Зарегистрироваться

[Войти в аккаунт](#)

Рисунок 5 – Форма регистрации

Авторизация

*e-mail

*Пароль

Войти

[Зарегистрироваться](#)

Рисунок 6 – Форма авторизации

После успешной авторизации пользователь перенаправляется на страницу «Мои фотографии», где отображены все фотографии пользователя (рисунок 7).

Photomarket Выйти

Мои фотографии
Мои альбомы
Создать фотографию
Создать альбом

Мои живые фотографии

<p>268 × 131</p> <p>Фотография 1</p> <p>01.04.2020</p> <p>Удалить</p>	<p>268 × 131</p> <p>Фотография 2</p> <p>01.04.2020</p> <p>Удалить</p>	<p>268 × 131</p> <p>Фотография 3</p> <p>01.04.2020</p> <p>Удалить</p>
<p>268 × 131</p> <p>Фотография 4</p> <p>01.04.2020</p> <p>Удалить</p>	<p>268 × 131</p> <p>Фотография 5</p> <p>01.04.2020</p> <p>Удалить</p>	

Рисунок 7 – Страница «Мои фотографии»

При помощи меню в левой верхней части экрана, пользователь может перемещаться по страницам сайта.

На странице «Создать фотографию» пользователь может создать новую живую фотографию. Для добавления фотографии и видео, их нужно перетащить в соответствующие поля, либо выбрать кликнув на них.

The screenshot shows the 'Создание живой фотографии' (Creating a live photo) interface. On the left is a sidebar menu with 'Мои фотографии', 'Мои альбомы', 'Создать фотографию', and 'Создать альбом'. The main area has a title bar 'Создание живой фотографии' and a 'Выйти' (Logout) button. Below the title is a text input field labeled '*Название фотографии'. There are two large rectangular boxes with blue upward-pointing arrows; the left one is labeled 'Перетащите или выберите фотографию' and the right one 'Перетащите или выберите видеозапись'. At the bottom right is a blue 'Сохранить' (Save) button.

Рисунок 8 – Страница «Создать фотографию»

При нажатии на ссылку «Мои альбомы», пользователь попадает на страницу с альбомами, здесь же находится форма для добавления альбома по идентификатору (рисунок 9).

The screenshot shows the 'Мои альбомы' (My Albums) interface. The sidebar menu is the same as in Figure 8. The main area has a title bar 'Мои альбомы' and a 'Выйти' (Logout) button. Below the title bar is a section titled 'Добавить альбом по идентификатору' (Add album by ID). It contains a text input field labeled '*Идентификатор альбома' and a blue 'Добавить' (Add) button. Below this are three identical album cards. Each card has a 'Скопировать идентификатор' (Copy ID) button at the top, followed by 'Выпускной' (Graduation) and the date '01.04.2020'. At the bottom of each card is a 'Удалить' (Delete) button. A fourth 'Скопировать идентификатор' button is located below the first card.

Рисунок 9 – Страница «Мои альбомы»

На странице «Создать альбом» пользователь может добавить фотографии в новый альбом заполнив соответствующие поля и нажав кнопку добавить. Когда все нужные фотографии добавлены, можно сохранить альбом, указав его название в соответствующем поле (рисунок 10).

Рисунок 10 – Страница «Создать альбом»

После добавления фотографии, они отображаются под формой добавления (рисунок 11).

Рисунок 11 – Страница «Создать альбом» с добавленными фотографиями

На основе созданного макета приложения можно разработать удобные пользовательские интерфейсы учитывающие все требования.

3.4 Разработка веб-клиента

3.4.1 Приложение на React

В качестве бэкенда веб-приложения выступает API рассмотренного выше серверного компонента информационной системы. Фронтенд разработан при помощи библиотеки React.

React – это эффективная и гибкая JavaScript библиотека для создания пользовательских интерфейсов. Она позволяет собирать сложный UI из маленьких изолированных кусочков кода, называемых «компонентами».

3.4.2 JSX

При написании кода с использованием библиотеки React используется JSX. JSX – это препроцессор, который добавляет синтаксис XML к JavaScript. Возможно использование React и без JSX, но JSX делает React более элегантным.

JSX – синтаксис, похожий на XML / HTML, используемый в React, расширяет ECMAScript, так что XML / HTML-подобный текст может сосуществовать с кодом JavaScript / React. Синтаксис предназначен для использования препроцессорами, чтобы преобразовать HTML-подобный текст, найденный в файлах JavaScript, в стандартные объекты JavaScript.

В основном, используя JSX, можно писать сжатые структуры HTML / XML (например, DOM подобные древовидные структуры) в том же файле, что и код JavaScript, а затем Babel преобразует эти выражения в код JavaScript. В отличие от прошлого, вместо того, чтобы помещать JavaScript в HTML, JSX позволяет нам помещать HTML в JavaScript.

3.4.3 Компоненты

React разработан вокруг концепции многоразовых компонентов. Вы определяете небольшие компоненты, и объединяете их, чтобы сформировать более крупные компоненты.

Все компоненты, маленькие или большие, могут использоваться повторно, даже в разных проектах. Компонент React – в его простейшей форме – это обычная функция JavaScript.

Для упрощения и ускорения разработки существуют библиотеки готовых компонентов для React. Одной из самых популярных библиотек является Material-UI.

Material-UI – это набор компонентов React, который реализует Google Material Design.

Material Design – стиль графического дизайна интерфейсов программного обеспечения и приложений, разработанный компанией Google. Впервые представлен на конференции Google I/O 25 июня 2014 года. Стиль расширяет идею «карточек», появившуюся в Google Now, более широким применением строгих макетов, анимаций и переходов, отступов и эффектов глубины (света и тени). По идее графических дизайнеров Google, у приложений не должно быть острых углов, карточки должны переключаться между собой плавно и практически незаметно.

У каждого компонента из библиотеки Material-UI есть API, который позволяет настраивать компонент под свои нужды. Таким образом из готовых компонентов можно формировать свои компоненты, тем самым быстро и удобно создавать интерфейс приложения.

3.4.4 Взаимодействие с сервером

Для взаимодействия с API сервером необходимо использование HTTP. В настоящее время для отправки HTTP-запросов в Javascript существует современный и очень мощный метод `fetch()`. Fetch API предоставляет интерфейс для получения ресурсов (в том числе по сети).

У `fetch` API имеется несколько недостатков. Он требует дополнительного действия для получения JSON из ответа. Также он не отлавливает все ошибки. Например, ошибка 404 возвращается как обычный

ответ. Вам нужно самим проверять код ответа и разбираться с ошибками сети, которые были получены.

Таким образом приходится обрабатывать ошибки дважды. Для решения этих проблем и получения более цельного кода, можно воспользоваться библиотекой `axios.js` за что придется добавить еще одну внешнюю зависимость. В данном приложении библиотека `axios.js` была использована как средство для взаимодействия с сервером по HTTP.

`Axios` – это широко известная JavaScript-библиотека. Она представляет собой HTTP-клиент, основанный на промисах и предназначенный для браузеров и для `Node.js`.

3.4.5 Авторизация

При осуществлении авторизации, в случае её успеха, клиентское приложение в качестве ответа от сервера получает токен, который необходимо отправлять с запросом на каждый защищенный маршрут для валидации запроса.

Для хранения токена используется `localStorage` браузера.

`LocalStorage` это свойство глобального объекта браузера (`window`). К нему можно обращаться как `window.localStorage` или просто `localStorage`. `LocalStorage` нужен для хранения определенных данные между сессиями пользователя.

Сохранив токен в `localStorage`, приложение в любой нужный момент может получить его для того, чтобы добавить в запрос.

Когда пользователь выходит из учетной записи, данные о токене в `localStorage` очищаются.

3.4.6 Маршрутизация

В любом реальном веб-приложении нужны маршруты, и приложение `React` не исключение. Пользователь должен видеть, где он находится в приложении в любой момент времени. А видит он свое текущее местоположение в адресной строке браузера. Следовательно, приложение

должно уметь сопоставлять определённый URL с соответствующей ему страницей. Также должна работать история. То есть когда пользователь кликает на стрелку "Назад" в браузере, приложение должно направить нас на предыдущую страницу.

Сам по себе React не предоставляет такой возможности, это задача специальных библиотек. Как правило, используя API такой библиотеки мы подключаем компоненты страниц нашего приложения, сопоставляя их с определёнными путями. После этого, переходя с одной страницы на другую мы будем видеть в адресной строке, как изменяется текущий путь.

Одной из таких библиотек является react-router-dom. Эта библиотека популярна, довольно проста в использовании и обладает хорошей документацией. Она предоставляет такие возможности как:

- навигация по клику (компонент Link);
- перенаправление (компонент Redirect);
- маршрутизация (компонент Route);
- история (свойство history).

Эта библиотека была использована в разрабатываемом приложении.

В зависимости от того, авторизован пользователь или нет, ему доступны разные маршруты.

Для неавторизованного пользователя доступны маршруты:

- /signin (страница авторизации);
- /signup (страница регистрации).

Для авторизованного пользователя доступны маршруты:

- /user (страница с фотографиями пользователя);
- /albums (страница с альбомами пользователя);
- /addPhoto (страница создания живого фото);
- /addAlbum (страница создания альбома).

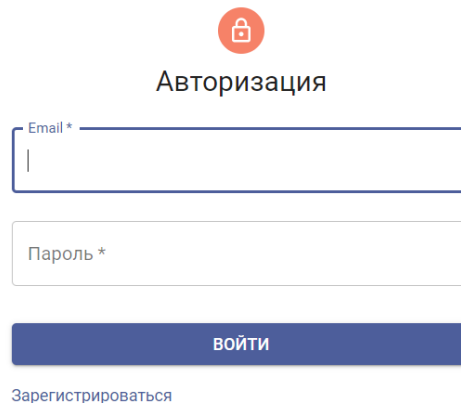
3.4.7 Реализация страниц приложения

С использованием компонентов библиотеки Material-UI, были созданы компоненты для всех необходимых страниц приложения.

При входе на сайт неавторизованного пользователя, открывается страница авторизации (рисунок 12). Здесь пользователь может пройти авторизацию, если у него есть зарегистрированная учетная запись, либо перейти на страницу регистрации (рисунок 13) и создать новую учетную запись.

На данных страницах реализуется функционал, соответствующий следующим требованиям:

- возможность авторизации;
- возможность регистрации.



Авторизация

Email *

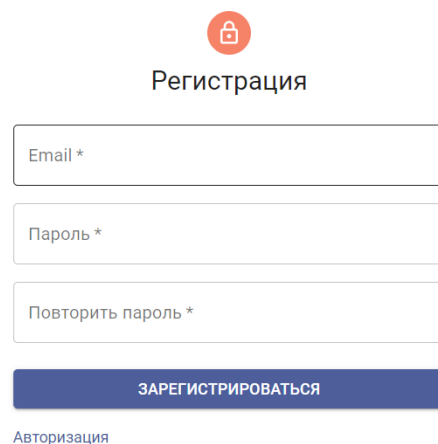
Пароль *

ВОЙТИ

[Зарегистрироваться](#)

Copyright © Developed by CS-solutions 2020.

Рисунок 12 – Страница авторизации



Registration form with the following fields and buttons:

- Email *
- Пароль *
- Повторить пароль *
- ЗАРЕГИСТРИРОВАТЬСЯ

Авторизация

Copyright © Developed by CS-solutions 2020.

Рисунок 13 – Страница регистрации

После успешной авторизации пользователь перенаправляется на страницу «Мои фотографии» (рисунок 14). На данной странице отображаются все личные фотографии пользователя, добавленные в систему в виде карточек.

На данной странице реализуется функционал, соответствующий следующим требованиям:

- возможность просмотреть добавленные в систему исходники (фото-видео);
- возможность удалить исходник.

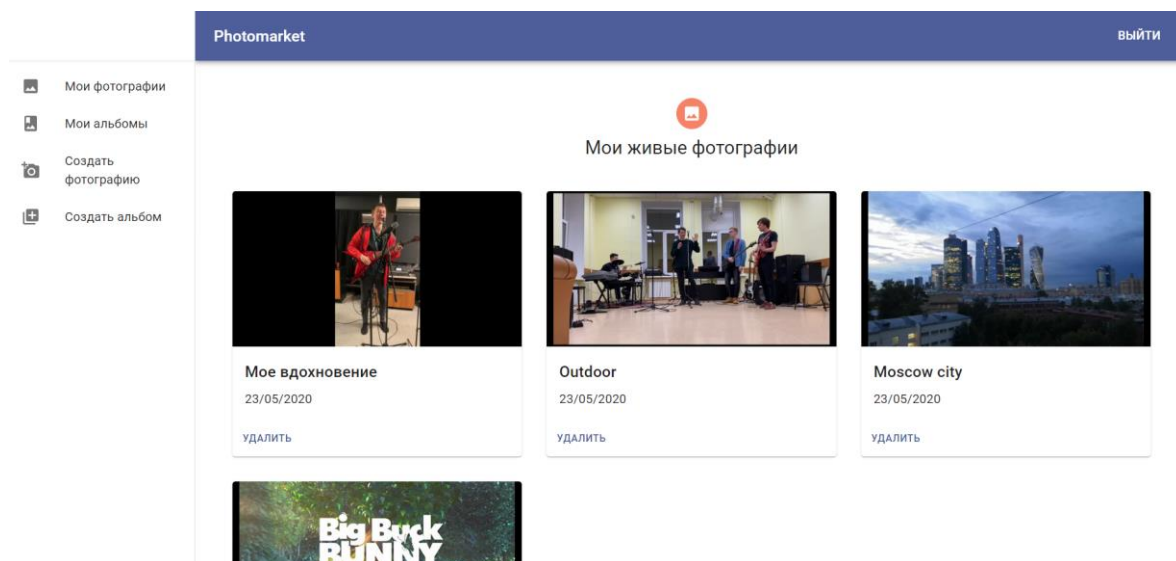


Рисунок 14 – Страница «Мои фотографии»

На каждой странице авторизованного пользователя присутствует панель навигации, позволяющая переходить между страницами (рисунок 15).

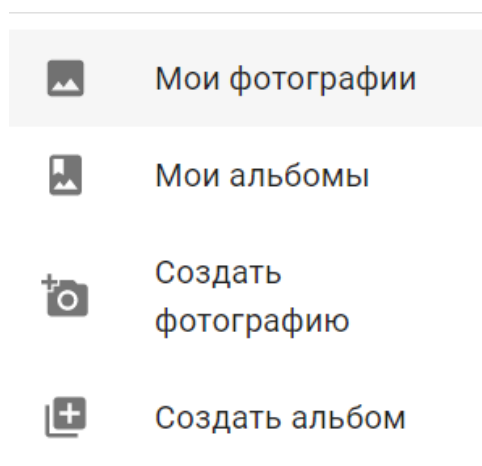


Рисунок 15 – Панель навигации

В карточке исходника (рисунок 16) находится кнопка «Удалить», которая позволяет удалить исходник из системы. При наведении курсора на фотографию в карточке, на ее месте запускается видеозапись из исходника.

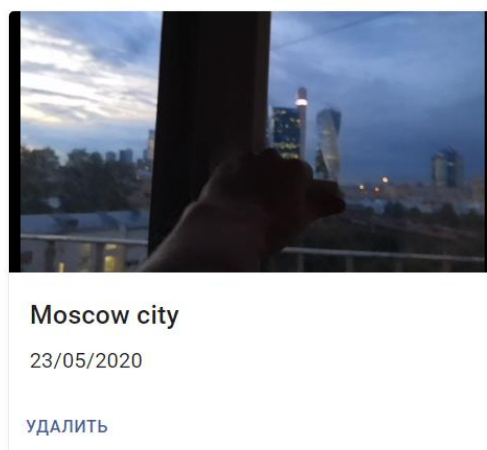


Рисунок 16 – Карточка исходника

На странице «Мои альбомы» (рисунок 17) пользователь может просмотреть все альбомы, которые привязаны к его учетной записи. Здесь же он может удалить альбом, тем самым отвязав все исходники из альбома, от своей учетной записи.

На этой же странице пользователь может получить идентификатор альбома, для того, чтобы передать его другому пользователю любым удобным способом (например, при помощи мессенджера). Пользователь получивший

идентификатор альбома, может привязать альбом к своей учетной записи, вставив идентификатор в соответствующее поле и нажав кнопку добавить.

На данной странице реализуется функционал, соответствующий следующим требованиям:

- возможность просмотреть добавленные альбомы;
- возможность удалить альбом;
- возможность поделиться альбомом.

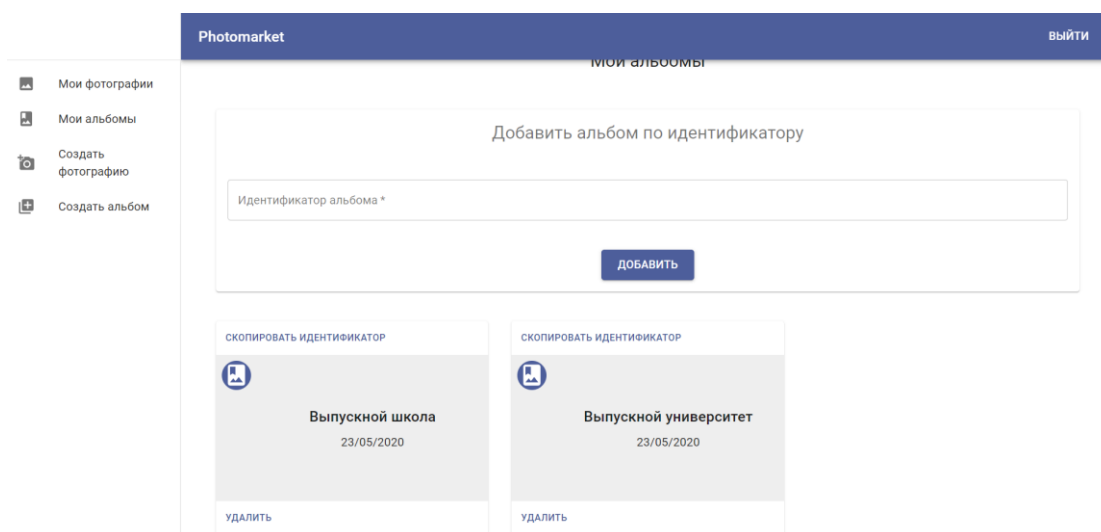


Рисунок 17 – Страница «Мои альбомы»

На странице «Создать фотографию» (рисунок 18), пользователь может добавить новый исходник в систему.

На данной странице реализуется функционал, соответствующий следующим требованиям:

- возможность добавить новый исходник (фото-видео).

The screenshot shows the 'Создание живой фотографии' (Create Live Photo) page. At the top is a dark blue header with 'Photomarket' on the left and 'ВЫЙТИ' (Logout) on the right. A left sidebar contains four items: 'Мои фотографии' (My Photos), 'Мои альбомы' (My Albums), 'Создать фотографию' (Create Photo), and 'Создать альбом' (Create Album). The main content area has a red camera icon and the title 'Создание живой фотографии'. Below the title is a text input field labeled 'Название фотографии *'. Underneath are two large dashed boxes: the left one says 'Перетащите или выберите фотографию' (Drag or select a photo) and the right one says 'Перетащите или выберите видео' (Drag or select a video), both with an upload icon. A blue 'СОХРАНИТЬ' (Save) button is at the bottom right.

Рисунок 18 – Страница «Создать фотографию»

На странице «Создать альбом» (рисунок 19), пользователь может создать новый альбом. Для этого ему нужно добавить исходники в альбом, для добавления исходника пользователю нужно заполнить соответствующие поля (рисунок 20) и нажать кнопку «Добавить». После добавления исходника, он отображается в виде карточки (рисунок 21). Когда все необходимые исходники добавлены, можно создать альбом, заполнив поле «Название альбома» и нажав кнопку «Сохранить».

На данной странице реализуется функционал, соответствующий следующим требованиям:

- возможность создания альбома.

The screenshot shows the 'Создание альбома живых фотографий' (Create Album of Live Photos) page. The header and sidebar are identical to the previous page. The main content area has the title 'Создание альбома живых фотографий'. It features a text input field for 'Название фотографии *'. Below it are two dashed boxes for adding content: 'Перетащите или выберите фотографию' and 'Перетащите или выберите видео', each with an upload icon. A blue 'ДОБАВИТЬ' (Add) button is at the bottom right of this section. At the bottom of the page is another text input field for 'Название альбома *' and a blue 'СОХРАНИТЬ' (Save) button.

Рисунок 19 – Страница «Создать альбом»

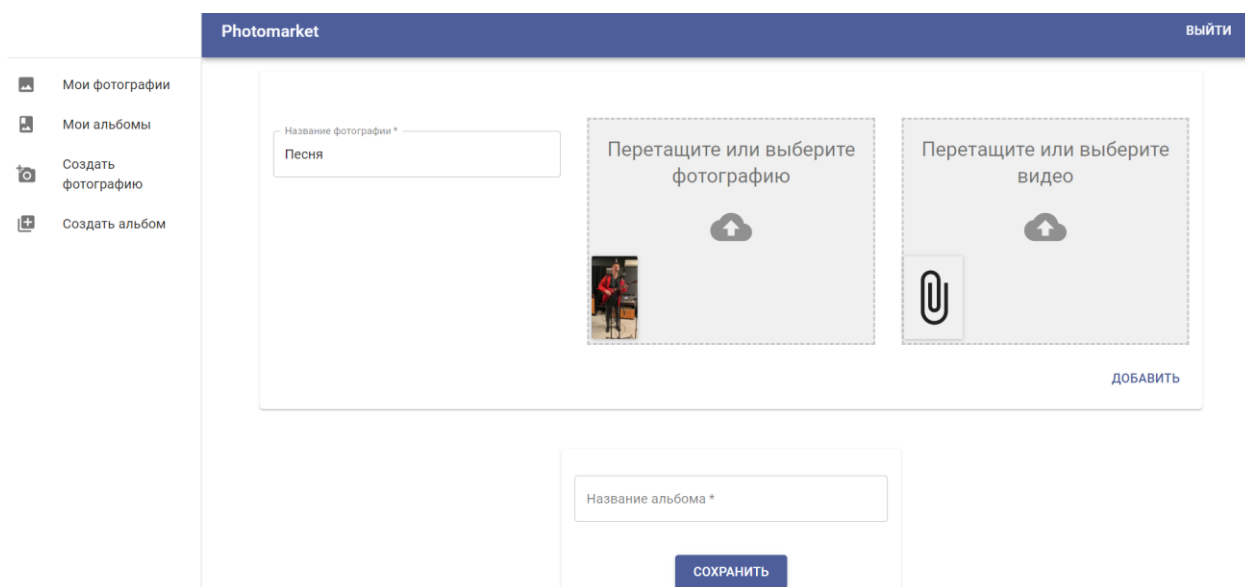


Рисунок 20 – Заполнены поля для создания исходника

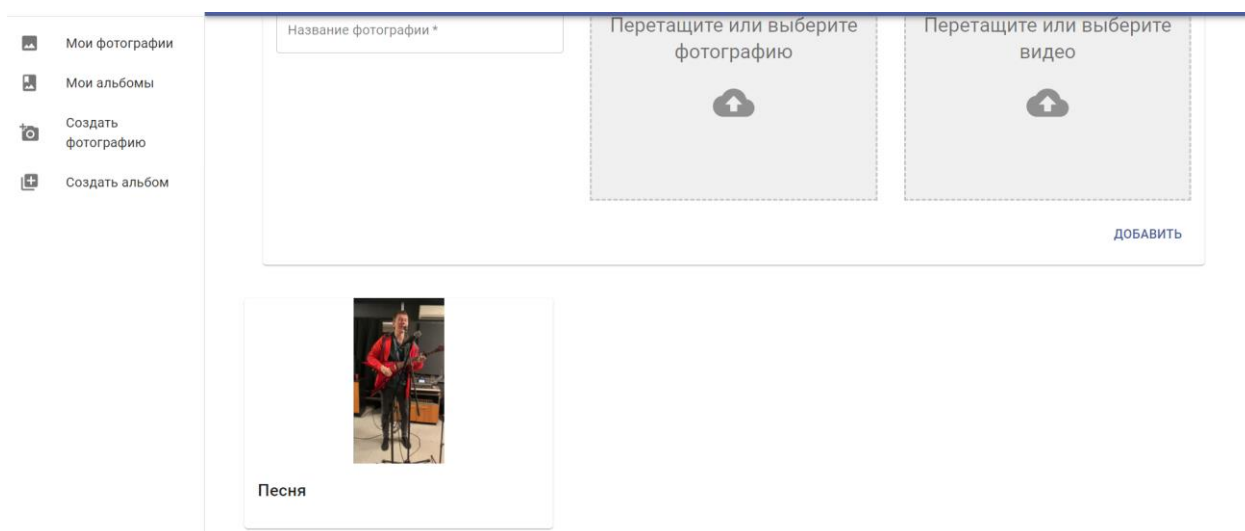


Рисунок 21 – Добавлен исходник

На каждой странице авторизованного пользователя присутствует кнопка «Выйти», которая позволяет выйти из учетной записи, таким образом реализуется требование «Возможность выхода из учетной записи».

Благодаря использованию отзывчивой верстки, приложение адаптировано под мобильные устройства. Пример приведен на рисунке 22.

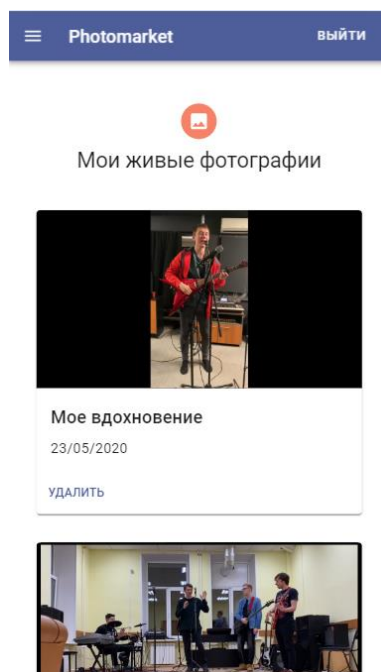


Рисунок 22 – Страница «Мои альбомы» с мобильного устройства

4 ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ

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

4.1.1 Потенциальные потребители результатов исследования

В рамках данной работы выполняется исследование, проектирование и разработка системы, обеспечивающей эффект дополненной реальности на изображении с камеры мобильного устройства. При этом объем информации, передаваемый обычной фотографией или стендом может быть увеличен за счет добавляемых объектов – текста, изображений, видеороликов, аудиозаписей. Таким образом, может быть сокращено использование традиционно используемых ресурсов – бумаги и пластика. Потенциальными потребителями являются университеты, музеи, школы, рекламные агентства. Особый интерес разработка представляет для организаций, занимающихся изготовлением памятной фото-продукции и сувениров.

4.1.2 Анализ конкурентных технических решений

При рассмотрении сферы изготовления фото-продукции (портретов, виньеток, коллажей), в качестве конкурентов можно выделить существующие фотоателье (К1) и компании, предоставляющие решения с использованием дополненной реальности (К2). Анализ конкурентных технических решений был проведен с помощью оценочной карты, представленной в таблице 7.

Таблица 7. Оценочная карта критериев эффективности конкурентных технических решений (разработок)

Критерии оценки	Вес критерия	Баллы			Конкурентоспособность		
		Б _ф	Б _{к1}	Б _{к2}	К _ф	К _{к1}	К _{к2}
Технические критерии оценки ресурсоэффективности							
Удобство в эксплуатации	0,1	5	5	5	0,5	0,5	0,5

Надежность	0,1	4	5	4	0,4	0,5	0,4
Инновационная привлекательность	0,15	5	2	5	0,75	0,3	0,75
Функциональная мощность	0,2	5	3	5	1	0,6	1
Простота в эксплуатации	0,1	5	5	4	0,5	0,5	0,4
Экономические критерии оценки эффективности							
Цена	0,15	4	5	3	0,6	0,75	0,45
Послепродажное обслуживание	0,1	4	5	2	0,4	0,5	0,2
Предполагаемый срок эксплуатации	0,1	4	4	4	0,4	0,4	0,4
Итого	1	36	34	32	4,55	4,05	4,1

Анализ конкурентных решений определяется по формуле 1:

$$K = \sum B_i * B_i \quad (1)$$

, где K – конкурентоспособность научной разработки или конкурента;

B_i – вес показателя (в долях единицы);

B_i – балл i-го показателя.

Основываясь на рассчитанных значениях, можно предположить, что инновационность разработки и расширение функциональных возможностей продукта в сочетании с простотой использования формирует конкурентное преимущество. Существующим альтернативам не хватает функциональной мощности в одном случае (фотоателье) и нацеленности на данную сферу в другом случае (организации, специализирующиеся на дополненной реальности).

4.1.3 Технология QuaD

Технология QuaD (QQuality ADvisor) представляет собой гибкий инструмент измерения характеристик, описывающих качество новой разработки и ее перспективность на рынке и позволяющие принимать решение целесообразности вложения денежных средств в научно-исследовательский проект.

Анализ имеющихся конкурентных продуктов необходимо проводить с достаточной регулярностью, так как рынок ИТ находится в постоянном и активном движении на сегодняшний день. Данный анализ позволяет производить изменения текущего внедрения, чтобы сделать его наиболее перспективным и конкурентоспособным в дальнейшем. Результат QuaD-анализа представлен в таблице 8.

Таблица 8. Оценка критериев в соответствии с технологией QuaD

Критерии оценки	Вес критерия	Баллы	Максимальный балл	Относительное значение	Средневзвешенное значение
1	2	3	4	5	6
Удобство в эксплуатации	0,1	100	100	1	0,1
Надежность	0,1	80	100	0,8	0,08
Инновационная привлекательность	0,15	100	100	1	0,15
Функциональная мощность	0,2	100	100	1	0,2
Простота в эксплуатации	0,1	100	100	1	0,1
Цена	0,15	80	100	0,8	0,12
Послепродажное обслуживание	0,1	80	100	0,8	0,08
Предполагаемый срок эксплуатации	0,1	80	100	0,8	0,08
Итого:					0,91

Средневзвешенное значение показателя качества и перспективности научной разработки равен 0,91 (попадает в промежуток 0.8 – 1), то есть такая разработка считается перспективной.

4.1.4 SWOT-анализ

В рамках данного анализа выявлены сильные и слабые стороны проекта, а также его возможности и угрозы. Результат SWOT-анализа представлен в таблице 9.

Таблица 9 – SWOT-анализ проекта

		Внутренние факторы	
		Сильные стороны проекта: 1. Простота интеграции 2. Уникальность и инновационность 3. Многофункциональность 4. Гибкость и простота внесения изменений	Слабые стороны проекта: 1. Необходимость длительной поддержки 2. Увеличение стоимости конечного продукта
Внешние факторы	Возможности: 1. Интеграция в процессы существующих организаций 2. Налаживание постоянного потока заказов от крупных организаций 3. Материальное поощрение от государства за деятельность в сфере экологии и культуры	Проект может стать востребованным как среди конечных потребителей так и среди крупных организаций	Увеличение стоимости конечного продукта с использованием предлагаемой разработки может произвести негативное впечатление как на конечного потребителя так и на посреднические организации
	Угрозы: 1. Несоответствие ожиданиям клиента 2. Проблемы с сертификацией деятельности 3. Некорректная работа ПО на некоторых устройствах	Благодаря возможности легкой модификации ПО, неожиданное несоответствие различным требованиям может быть устранено в кратчайшие сроки	Издержки на исправления несоответствий в сочетании с добавочной стоимостью могут сделать решение нерентабельным

SWOT-анализ показал, что риски, связанные с устранением несоответствий на всех этапах эксплуатации продукта, должны занимать ключевую роль в расчете выгоды предлагаемой разработки. Общая картина говорит о том, что, несмотря на угрозы и слабые стороны проекта, разработка обладает конкурентным преимуществом и является перспективной.

4.2 Планирование научно-исследовательских работ

4.2.1 Структура работ в рамках научного исследования

Важным этапом проведения научно-исследовательских работ является необходимость планирования работ, которое включает в себя определение полного перечня работ, а также их распределение между всеми исполнителями проекта. Исполнителями проекта являются студент и научный руководитель. Научный руководитель определяет цели и задачи для студента, направляет и контролирует работу его работу, оценивает результаты проделанной работы и дает рекомендации студенту. Студент полностью отвечает за выполняемую работу. В таблице 10 представлен перечень работ, а также распределение исполнителей по ним.

Таблица 10. Перечень работ и распределение исполнителей

№ работы	Наименование работы	Исполнители работы
1	Выбор научного руководителя бакалаврской работы	Козловский В.Е., Степаненко В.В.
2	Составление и утверждение темы бакалаврской работы	Козловский В.Е., Степаненко В.В., Цапко С.Г.
3	Составление календарного плана-графика выполнения бакалаврской работы	Цапко С.Г.
4	Подбор и изучение литературы по теме бакалаврской работы	Козловский В.Е., Степаненко В.В., Цапко С.Г.
5	Анализ предметной области	Козловский В.Е., Степаненко В.В., Цапко С.Г.

6	Проектирование информационной системы	Козловский В.Е., Степаненко В.В., Цапко С.Г.
7	Разработка клиентского приложения системы	Козловский В.Е.
8	Разработка сервера системы	Степаненко В.В.
9	Согласование выполненной работы с научным руководителем	Козловский В.Е., Степаненко В.В., Цапко С.Г.
10	Выполнение других частей работы (финансовый менеджмент, социальная ответственность)	Козловский В.Е., Степаненко В.В.
11	Подведение итогов, оформление работы	Козловский В.Е., Степаненко В.В.

4.2.2 Определение трудоемкости выполнения работ

Как правило, трудовые затраты образуют основную часть стоимости исследования, поэтому важным этапом планирования научно-исследовательской деятельности является определение трудоемкости работ.

Определение трудоёмкости выполнения работ осуществляется на основе экспертной оценки ожидаемой трудоёмкости выполнения каждой работы путём расчёта длительности работ в рабочих и календарных днях каждого этапа работ.

Трудоемкость оценивается по следующей формуле 2:

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

где $t_{\text{ож } i}$ – это ожидаемая трудоемкость i -ой работы (чел.-дни),

$t_{\text{min } i}$ – это минимально возможная трудоемкость выполнения заданной i -ой работы (оптимистическая оценка) (чел.-дни),

$t_{\text{max } i}$ – это максимально возможная трудоемкость выполнения заданной i -ой работы (пессимистическая оценка) (чел.-дни).

После оценки ожидаемой трудоемкости работ, производится определение продолжительности каждой работы в рабочих днях по формуле 3:

$$T_{pi} = \frac{t_{ож i}}{Ч_i}, \quad (3)$$

где T_{pi} – это продолжительность одной работы (раб. дни),

$t_{ож i}$ – это ожидаемая трудоемкость выполнения одной работы (чел.-дни),

$Ч_i$ – это численность исполнителей, выполняющих одновременно одну и ту же работу на этом этапе (чел.).

Для того чтобы в дальнейшем построить график работ с помощью диаграммы Ганта, необходимо также произвести перевод длительности работ из рабочих дней в календарные по формуле 4:

$$T_{ki} = T_{pi} \cdot k_{кал}, \quad (4)$$

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

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

$k_{кал}$ – это коэффициент календарности, равный 1,22.

Коэффициент календарности $k_{кал}$ рассчитывается по формуле 5:

$$k_{кал} = \frac{T_{кал}}{T_{кал} - T_{вых} - T_{пр}}, \quad (5)$$

где $k_{кал}$ – коэффициент календарности;

$T_{кал}$ – количество календарных дней в году;

$T_{кал}$ – количество выходных дней в году;

$T_{пр}$ – количество праздничных дней в году.

С учётом того, что календарных дней в 2020 году 366, а сумма выходных и праздничных дней составляет 69 дней, коэффициент календарности равен $k_{кал} = 1,23$.

4.2.3 Разработка графика проведения научного исследования

В таблице 11 приведены временные показатели научного исследования.

Таблица 11. Временные показатели проведения научного исследования

Наименование работы	Исполнители работы	Трудоемкость работ, чел-дни		Длительность работ, дни		
		tmin	tmax	toж	Tr	Tк
Выбор научного руководителя бакалаврской работы	Козловский В.Е. Степаненко В.В.	1	1	1	1	1
Составление и утверждение темы бакалаврской работы	Козловский В.Е. Степаненко В.В.	1	3	1,8	1	1
Составление календарного плана-графика выполнения бакалаврской работы	Цапко С.Г.	1	1	1	1	1
Подбор и изучение литературы по теме бакалаврской работы	Козловский В.Е. Степаненко В.В.	5	10	7	4	5
	Цапко С.Г.	1	1	1	1	1
Анализ предметной области	Козловский В.Е. Степаненко В.В.	5	10	7	4	5
	Цапко С.Г.	1	2	1,4	1	1
Проектирование информационной системы	Козловский В.Е. Степаненко В.В.	10	20	14	7	9
	Цапко С.Г.	1	2	1,4	1	1
Разработка клиентского приложения системы	Козловский В.Е.	20	40	28	28	34
Разработка сервера системы	Степаненко В.В.	20	40	28	28	34
Согласование выполненной работы с научным руководителем	Козловский В.Е. Степаненко В.В.	1	1	1	1	1
	Цапко С.Г.	1	2	1,4	1	1
Выполнение других частей работы (финансовый менеджмент, социальная ответственность)	Козловский В.Е., Степаненко В.В.	10	14	11,6	12	15
Подведение итогов, оформление работы	Козловский В.Е., Степаненко В.В.	5	7	5,8	6	7
Итого	Козловский В.Е.	58	106	77,2	77	95
	Степаненко В.В.	58	106	77,2	77	95
	Цапко С.Г.	5	8	6,2	6	7

На основе данных таблицы 5 был построен календарный план-график (рисунок 23).

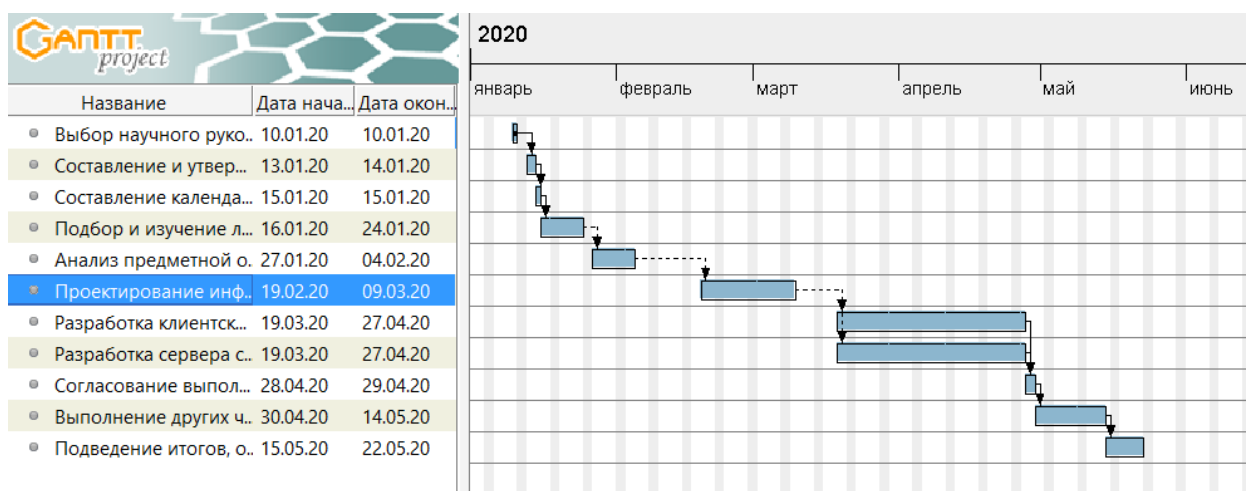


Рисунок 23 – Календарный план-график проведения работ

4.2.4 Бюджет научно-технического исследования

В состав бюджета входит стоимость всех расходов, необходимых для выполнения работ по проекту. При формировании бюджета используется группировка затрат по следующим статьям:

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

4.2.4.1 Расчет материальных затрат научно-технического исследования

Данная статья затрат включает в себя затраты на приобретение сырья, материалов, полуфабрикатов и комплектующих со стороны, используемых при разработке проекта. Также в эту статью включаются транспортные расходы, равные 15% от общей стоимости материальных затрат.

Затраты на канцелярские принадлежности составляют 500 руб.

Таким образом, сумма материальных затрат составляет 500 руб.

4.2.4.2 Расчет затрат на специальное оборудование для научных (экспериментальных) целей

Данная статья затрат включает в себя затраты на приобретение специального оборудования. Также в эту статью включаются затраты по доставке и монтажу оборудования, равные 15% от его стоимости.

В ходе работы над проектом использовалось оборудование, имеющееся у исполнителей, соответственно необходим расчет его амортизации.

При создании информационной системы были использованы два персональных компьютера, стоимость 40000 и 60000 рублей, суммарная стоимость 100000 рублей.

Расчет амортизации ПК: первоначальная стоимость ПК 100000 рублей; срок полезного использования для машин офисных код 330.28.23.23 составляет 36 месяцев. Планируемое время использования ПК для написания ВКР - 6 месяцев.

Норма амортизация основных средств линейным способом рассчитывается по формуле 6:

$$A_n = \frac{1}{n} * 100\% , \quad (6)$$

n – установленный срок в месяцах;

A_n – норма амортизации.

Тогда расчет амортизации ПК:

– норма амортизации:

$$A_n = \frac{1}{n} * 100\% = \frac{1}{36} \times 100\% = 2,78\%$$

– ежемесячные амортизационные отчисления:

$$A_m = 100000 * 0,0278 = 2780 \text{ рублей}$$

– итоговая сумма амортизации основных средств:

$$A = 2780 \times 6 = 16680 \text{ рублей}$$

Таким образом, сумма затрат на специальное оборудование составляет 16680 рублей, в виде амортизационных отчислений.

4.2.4.3 Основная заработная плата исполнителей темы

Данная статья затрат включает основную заработную плату, премии и доплаты всех исполнителей проекта. В качестве исполнителей проекта выступают студенты и научный руководитель.

Заработная плата рассчитывается по формуле 7:

$$З_{зп} = З_{осн} + З_{доп}, \quad (7)$$

где $З_{зп}$ – заработная плата исполнителя;

$З_{осн}$ – основная заработная плата исполнителя;

$З_{доп}$ – дополнительная заработная плата исполнителя (12%-15% от размера основной заработной платы).

Основную заработную плату можно получить по формуле 8:

$$З_{осн} = З_{дн} \times Тр \times (1 + Кпр + Кд) \times Кр \quad (8)$$

где $З_{дн}$ – среднедневная заработная плата, руб.

$Кпр$ – премиальный коэффициент (0,3);

$Кд$ – коэффициент доплат и надбавок (0,2-0,5);

$Кр$ – районный коэффициент (для Томска 1,3);

$Тр$ – продолжительность работ, выполняемых работником, раб. дни

где $З_{дн}$ – среднедневная заработная плата исполнителя;

$Тр$ – продолжительность работ, выполняемых исполнителем.

Среднедневную заработную плату можно получить по формуле 9:

$$З_{дн} = \frac{З_{м*М}}{F_{д}}, \quad (9)$$

где $З_{м}$ – месячный должностной оклад исполнителя, рубли;

$М$ – количество месяцев работы равно:

При отпуске в 24 рабочих дня $М = 11,2$ месяца, 5 – дневная неделя;

При отпуске в 48 рабочих дней $М = 10,4$ месяца, 6 дневная неделя;

$F_{д}$ – действительный годовой фонд рабочего времени персонала по разработке.

Должностные оклады исполнителей проекта согласно приказу ТПУ представлены в таблице 12.

Таблица 12 – Месячные должностные оклады исполнителей

Исполнитель	Районный коэффициент (для Томска)	Размер месячного должностного оклада без учета коэффициента, рубли
Научный руководитель (должность – доцент, степень – кандидат технических наук)	1,3	33664
Студент (ассистент, без степени)	1,3	21760

Баланс рабочего времени для 6-дневной рабочей недели представлен в таблице 13.

Таблица 13 – Баланс рабочего времени (для 6-дневной недели)

Показатели рабочего времени	Дни
Календарные дни	366
Нерабочие дни (праздники/выходные)	66
Потери рабочего времени (отпуск/невыходы по болезни)	56
Действительный годовой фонд рабочего времени	244

На основе формулы 8 и таблиц 12–13 была рассчитана среднедневная заработная плата:

$$З_{\text{дн}}(\text{студент}) = \frac{21760 * 10,4}{244} = 927,48 \text{ рубля}$$

$$З_{\text{дн}}(\text{научный руководитель}) = \frac{33664 * 10,4}{243} = 1434,86 \text{ рублей}$$

Расчет затрат на основную заработную плату приведен в таблице 14.

Таблица 14 – Затраты на основную заработную плату

Исполнители	Здн, руб.	Кпр	Кд	Кр	Тр	Зосн, руб.
-------------	-----------	-----	----	----	----	------------

Студент 1	927,48	0,3	0,2	1,3	77	139261,12
Студент 2	927,48	0,3	0,2	1,3	77	139261,12
Научный руководитель	1434,86	0,3	0,2	1,3	6	16787,86
Итого:						295310,10

Итоговая сумма затрат на основную заработную плату составила 295310,10 руб.

4.2.4.4 Дополнительная заработная плата исполнителей темы

Данная статья расходов учитывает величину предусмотренных Трудовым кодексом РФ доплат за отклонение от нормальных условий труда и выплат, связанных с обеспечением гарантий и компенсаций.

Расчёт дополнительной заработной платы осуществляется по формуле 10:

$$З_{\text{доп}} = k_{\text{доп}} \times З_{\text{осн}}, \quad (10)$$

где $З_{\text{доп}}$ – дополнительная заработная плата, рубли;

$k_{\text{доп}}$ – коэффициент дополнительной заработной платы (на стадии проектирования принимается равным 0,12 – 0,15);

$З_{\text{осн}}$ – основная заработная плата, рубли.

Расчет затрат на дополнительную заработную плату приведен в таблице 15.

Таблица 15 – Затраты на дополнительную заработную плату

Исполнители	Зосн, руб.	Кдоп	Здоп, руб.
Студент 1	139261,12	0,12	16711,33
Студент 2	139261,12	0,12	16711,33
Научный руководитель	16787,86	0,12	2014,54
Итого:			35437,20

Итоговая сумма затрат на дополнительную заработную плату составила 35437,20 руб.

4.2.4.5 Отчисления во внебюджетные фонды (страховые отчисления)

К отчислениям во внебюджетные фонды относятся отчисления:

- отчисления органам государственного социального страхования (ФСС);
- отчисления в пенсионный фонд (ПФ);
- отчисления медицинского страхования (ФФОМС).

Сумма отчислений во внебюджетные фонды рассчитывается на основе затрат на оплату труда исполнителей и может быть вычислена по формуле 11.

$$З_{внеб} = k_{внеб} * (З_{осн} + З_{доп}), \quad (11)$$

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

Размер коэффициента определяется законодательно и в настоящее время согласно Федеральному закону от 24.07.2009 №212-ФЗ установлен в размере 30%.

Расчет затрат на отчисления во внебюджетные фонды приведен в таблице 16.

Таблица 16 – Отчисления во внебюджетные фонды

Исполнители	Зосн, руб.	Здоп, руб.	Квнеб	Звнеб, руб.
Студент	139261,12	16711,33	0,3	46791,74
	139261,12	16711,33	0,3	46791,74
Научный руководитель	16787,86	2014,54	0,3	5640,72
Итого:				99224,20

Итоговая сумма отчислений во внебюджетные фонды составила 99224,20 руб.

4.2.4.6 Накладные расходы

Накладные расходы – расходы на организацию, управление и обслуживание процесса производства товара, оказания услуги; носят комплексный характер. Накладные расходы вычисляются по формуле 12:

$$Z_{\text{нкл}} = \text{сумма статей} (1 - 5) * k_{\text{нр}}, \quad (12)$$

где $k_{\text{нр}}$ – коэффициент накладных расходов (16% от суммы затрат, подсчитанных выше).

Расчет накладных расходов приведен в таблице 17.

Таблица 17 – Расчет накладных расходов

Статьи затрат	Сумма, руб.
Материальные затраты	500
Затраты на специальное оборудование	16680
Затраты на основную заработную плату	295310,10
Затраты на дополнительную заработную плату исполнителям проекта	35437,20
Затраты на отчисления во внебюджетные фонды	99224,20
Коэффициент накладных расходов	0,16
Накладные расходы	71544,24

Итоговая сумма накладных расходов составила 71544,24 руб.

4.2.4.7 Формирование бюджета затрат научно-исследовательского проекта

После того, как была подсчитана каждая из статей расходов, можно приступить к формированию общего бюджета затрат проекта. Итоговый бюджет затрат представлен в таблице 18.

Таблица 18 – Расчет бюджета затрат НТИ

Наименование статьи	Сумма, руб.	Удельный вес, %	Примечание
Материальные затраты	500	0,8	Пункт 2.4.1
Затраты на специальное оборудование	16680	2,1	Пункт 2.4.2
Затраты на основную	295310,10	57,2	Пункт 2.4.3

заработную плату			
Затраты на дополнительную заработную плату	35437,20	6,9	Пункт 2.4.4
Страховые взносы	99224,20	19,2	Пункт 2.4.5
Накладные расходы	71544,24	13,8	Пункт 2.4.6
Общий бюджет	518695,74	100	Сумма ст. 1-6

Таким образом, общий бюджет НТИ составляет 518695,74 рубля.

4.3 Определение потенциального эффекта исследования

В рамках данного раздела был проведен комплексный SWOT-анализ проекта, который позволил выявить его сильные и слабые стороны, а также определить соответствие его характеристик внешним факторам. Более того, были выявлены возможные пути дальнейшего развития и совершенствования проекта для повышения его конкурентоспособности. Также была произведена оценка качества и перспективности данного проекта.

Еще одним ключевым моментом, рассмотренным в данном разделе, является планирование работ, выполняемых в рамках проекта. Составленный список необходимых работ с распределением исполнителей, а также вычисленная трудоемкость, позволили построить наглядный план-график работ в виде диаграммы Ганта. Это позволило рационально распорядиться временными ресурсами, отведенными на выполнение проекта.

Кроме того, был определен бюджет проекта. Определены затраты на заработные платы исполнителям с учетом страховых отчислений, рассчитаны материальные затраты, накладные расходы, а также амортизация оборудования, задействованного в процессе выполнения проекта.

5 СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ

5.1 Введение

Целью данной выпускной квалификационной работы является разработка серверного программного обеспечения и клиентского веб-приложения для информационной системы с технологией интерактивной визуализации средствами дополненной реальности. Информационная система позволяет пользователю «оживлять» фотографии, при помощи камеры мобильного телефона, средствами дополненной реальности, заменяя изображения фотографии на экране телефона соответствующим видео. Также пользователю предоставляется удобный веб-интерфейс, позволяющий создавать пары фотография-видео.

В данном разделе ВКР рассматриваются вопросы, связанные с соблюдением правовых и организационных вопросов обеспечения безопасности при разработке и использовании описываемой системы. Так как разработка и использование информационной системы производится на месте оператора электронной вычислительной машины, то в данном разделе рассмотрены вредные и опасные факторы которые могут возникнуть при использовании средств вычислительной техники. Также в данном разделе рассмотрены вопросы охраны окружающей среды от негативного воздействия использования информационной системы и возможные чрезвычайные ситуации, которые могут возникнуть.

5.2 Правовые и организационные вопросы обеспечения безопасности

Регулирование отношений между работником и работодателем, касающихся оплаты труда, трудового распорядка, особенности регулирования труда женщин, детей, людей с ограниченными способностями и проч., осуществляется законодательством РФ, а именно трудовым кодексом РФ.

Нормальная продолжительность рабочего времени не может превышать 40 часов в неделю.

Режим труда и отдыха предусматривает соблюдение определенной длительности непрерывной работы на персональном компьютере (ПК) и перерывов, регламентированных с учетом продолжительности рабочей смены, видов и категории трудовой деятельности.

Вид трудовой деятельности на персональном компьютере в рамках данной работы соответствует группе В – творческая работа в режиме диалога с ПК, категория трудовой деятельности – III (до 6 часов непосредственной работы на ПК).

При 8-часовой рабочей смене и работе на ПК, соответствующей описанным выше критериям необходимо через 1,5- 2,0 часа от начала рабочей смены и через 1,5-2,0 часа после обеденного перерыва устраивать регламентированные перерывы продолжительностью 20 минут каждый или продолжительностью 15 минут через каждый час работы.

Продолжительность непрерывной работы на ПК без регламентированного перерыва не должна превышать 2 часа.

Эффективными являются нерегламентированные перерывы (микропаузы) длительностью 1-3 минуты для выполнения комплекса упражнений.

Регламентированные перерывы и микропаузы целесообразно использовать для выполнения комплекса упражнений и гимнастики для глаз, пальцев рук, а также массажа. Комплексы упражнений целесообразно менять через 2-3 недели.

Продолжительность рабочего дня не должна быть меньше указанного времени в договоре, но не больше 40 часов в неделю. Для работников до 16 лет – не более 24 часов в неделю, от 16 до 18 лет и инвалидов I и II группы – не более 35 часов.

Продолжительность работы (смены) в ночное время сокращается на один час без последующей отработки. К работе в ночное время не допускаются: беременные женщины; работники, не достигшие возраста 18 лет, за исключением лиц, участвующих в создании и (или) исполнении художественных произведений, и других категорий работников в соответствии с настоящим Кодексом и иными федеральными законами.

В течение рабочего дня (смены) работнику должен быть предоставлен перерыв для отдыха и питания. Время предоставления перерыва и его конкретная продолжительность устанавливаются правилами внутреннего трудового распорядка или по соглашению между работником и работодателем.

Всем работникам предоставляются выходные дни (еженедельный непрерывный отдых).

Законодательством РФ запрещена дискриминация по любым признакам и принудительный труд [8].

Если пользователь постоянно загружен работой с ЭВМ, приемлемой является поза сидя. В положении сидя основная нагрузка падает на мышцы, поддерживающие позвоночный столб и голову. В связи с этим при длительном сидении время от времени необходимо сменять фиксированные рабочие позы.

Исходя из общих принципов организации рабочего места, в нормативно-методических документах сформулированы требования к конструкции рабочего места.

Рабочее место программиста должно соответствовать требованиям СанПин 2.2.2/2.4.1340-03.

Взаимное расположение элементов рабочего места должно обеспечивать возможность осуществления всех необходимых движений и

перемещений для эксплуатации и технического обслуживания оборудования [9].

Рабочие места с ЭВМ должны располагаться на расстоянии не менее 1,5 м от стены с оконными проемами, от других стен – на расстоянии 1 м, между собой – на расстоянии не менее 1,5 м. При размещении рабочих мест необходимо исключить возможность прямой засветки экрана источником естественного освещения.

При размещении ЭВМ на рабочем месте должно обеспечиваться пространство для пользователя величиной не менее 850 мм. Для стоп должно быть предусмотрено пространство по глубине и высоте не менее 150 мм, по ширине – не менее 530 мм. Располагать ЭВМ на рабочем месте необходимо так, чтобы поверхность экрана находилась на расстоянии 400 – 700 мм от глаз пользователя. Конструкция рабочего места и взаимное расположение всех его элементов (сиденье, органы управления, средства отображения информации и т.д.) должны соответствовать антропометрическим, физиологическим и психологическим требованиям, а также характеру работы [10].

Высота поверхности сиденья должна регулироваться в пределах 400 – 550 мм. Ширина и глубина его поверхности должна быть не менее 400 мм. Поверхность сиденья должна быть плоской, передний край – закругленным. Сиденье и спинка кресла должны быть полумягкими, с нескользящим, не электризующимся и воздухопроницаемым покрытием, материал которого обеспечивает возможность легкой очистки от загрязнения.

Опорная поверхность спинки стула должна иметь высоту 280 – 320 мм, ширину – не менее 380 мм и радиус кривизны горизонтальной плоскости – 400 мм. Расстояние спинки от переднего края сиденья должно регулироваться в пределах 260 – 400 мм.

При организации рабочего пространства необходимо учитывать индивидуальные антропометрические параметры пользователя с соответствующими допусками на возможные изменения рабочих поз и потребность в перемещениях.

Рациональной рабочей позой может считаться такое расположение тела, при котором ступни работника расположены на плоскости пола или на подставке для ног, бедра сориентированы в горизонтальной плоскости, верхние части рук – вертикальный угол локтевого сустава колеблется в пределах 70 – 90, запястья согнуты под углом не более чем 20, наклон головы – в пределах 15 – 20, а также исключены частые ее повороты [11].

5.3 Производственная безопасность

Для обеспечения производственной безопасности необходимо проанализировать воздействия на человека вредных и опасных производственных факторов, которые могут возникать при разработке проекта.

На человека в процессе его трудовой деятельности могут воздействовать опасные (вызывающие травмы) и вредные (вызывающие заболевания) производственные факторы (ГОСТ 12.0.003-2015), которые разделяются на четыре группы: физические, химические, биологические и психофизиологические. [12]

Были выявлены опасные и вредные факторы, действующие на пользователя ПЭВМ. Выявленные факторы представлены в таблице 19.

Таблица 19 – Возможные опасные и вредные факторы действующие на пользователя ПЭВМ

Факторы	Этапы работы		Нормативные документы
	Разработка	Эксплуатация	
Отклонение показателей микроклимата в помещении	+	+	<ul style="list-style-type: none"> • СН 2.2.4/2.1.8.562-96 • СанПиН 2.2.4.548-96 • СанПиН 2.2.2/2.4.1340-03
Недостаточная освещенность рабочей зоны	+	+	<ul style="list-style-type: none"> • СП 52.13330.2011 • ГОСТ Р 12.1.019-2009 ССБТ

Умственное перенапряжение	+		• СНиП 21-01-97
Опасность поражения электрическим током	+	+	

5.4 Анализ выявленных вредных и опасных факторов

5.4.1 Отклонение показателей микроклимата в помещении

Микроклимат производственных помещений – это климат внутренней среды данных помещений, который определяется совместно действующими на организм человека температурой, относительной влажностью и скоростью движения воздуха, а также температурой окружающих поверхностей является одним из важнейших условий существования человека.

Факторы, влияющие на микроклимат, можно разделить на две группы: нерегулируемые (комплекс климатообразующих факторов данной местности) и регулируемые (особенности и качество строительства зданий и сооружений, интенсивность теплового излучения от нагревательных приборов, кратность воздухообмена, количество людей и животных в помещении и др.). Для поддержания параметров воздушной среды рабочих зон в пределах гигиенических норм решающее значение принадлежит факторам второй группы.

Показатели микроклимата должны обеспечивать сохранение теплового баланса человека с окружающей средой и поддержание оптимального или допустимого теплового состояния организма.

Показателями, характеризующими микроклимат в производственных помещениях, являются:

- температура воздуха;
- температура поверхностей;

- относительная влажность воздуха;
- скорость движения воздуха;
- интенсивность теплового облучения [13].

При длительном и систематическом пребывании человека в оптимальных микроклиматических условиях сохраняется нормальное функциональное и тепловое состояние организма без напряжения механизмов терморегуляции. При этом ощущается тепловой комфорт (состояние удовлетворения внешней средой), обеспечивается высокий уровень работоспособности. Такие условия предпочтительны на рабочих местах. Значения оптимальных величин для категории работ по уровню энергозатрат Ia приведены в таблице 20.

Таблица 20 – Оптимальные величины показателей микроклимата на рабочих местах производственных помещений

Период года	Температура воздуха, °С	Температура поверхностей, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с
Холодный	22-24	21-25	60-40	0,1
Теплый	23-25	22-26	60-40	0,1

Допустимые микроклиматические условия при длительном и систематическом воздействии на человека могут вызвать преходящие и быстро нормализующиеся изменения функционального и теплового состояния организма и напряжение механизмов терморегуляции, не выходящие за пределы физиологических приспособительных возможностей. При этом не нарушается состояние здоровья, но возможны дискомфортные теплоощущения, ухудшение самочувствия и снижение работоспособности. Значения допустимых величин для категории работ по уровню энергозатрат Ia приведены в таблице 21.

Таблица 21 – Допустимые величины показателей микроклимата на рабочих местах производственных помещений

Период года	Температура воздуха, °С		Температура поверхностей, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с	
	Ниже оптимальных	Выше оптимальных			Ниже оптимальных	Выше оптимальных
Холодный	20-21,9	24,1-25	19-26	15-75	0,1	0,1
Теплый	21-22,9	25,1-28	20-29	15-75	0,1	0,2

5.4.2 Недостаточная освещенность рабочей зоны

Свет является одним из важнейших условий существования человека.

Недостаточная освещенность рабочей зоны является вредным производственным фактором, который может вызвать ослепленность или привести к быстрому утомлению и снижению работоспособности.

Причиной недостаточной освещенности являются недостаточность естественного освещения, недостаточность искусственного освещения, пониженная контрастность.

Работа с компьютером подразумевает постоянный зрительный контакт с дисплеем ПЭВМ и занимает от 80 % рабочего времени. Недостаточность освещения снижает производительность труда, увеличивает утомляемость и количество допускаемых ошибок, а также может привести к появлению профессиональных болезней зрения.

Разряд зрительных работ специалиста по внедрению и оператора ПЭВМ относится к разряду III и подразряду Г (работы высокой точности).

Для создания и поддержания благоприятных условий освещения для операторов ПЭВМ, их рабочие места должны соответствовать санитарно-эпидемиологическим правилам СанПиН 2.2.2/2.4.1340-03. Рабочее помещение должно иметь естественное и искусственное освещение. Для рассеивания

естественного освещения следует использовать жалюзи на окнах рабочих помещений. В качестве источников искусственного освещения должны быть использованы люминесцентные лампы, лампы накаливания – для местного освещения [14].

5.4.3 Умственное перенапряжение

Наличие большого количества потребляемой и анализируемой информации на рабочем месте, что характерно для оператора ПЭВМ, может происходить умственное перенапряжение.

Умственное переутомление особенно опасно для психического здоровья человека, оно связано со способностью центральной нервной системы долго работать с перегрузками, а это в конечном итоге может привести к развитию запредельного торможения, к нарушению сна, разлаженности взаимодействия вегетативных функций. Для профилактики умственного перенапряжения необходимо выполнять перерывы в работе.

5.4.4 Опасность поражения электрическим током

Поражение электрическим током является опасным производственным фактором и, поскольку специалист имеет дело с электрооборудованием, то вопросам электробезопасности на его рабочем месте должно уделяться особое внимание. Нормы электробезопасности на рабочем месте регламентируются СанПиН 2.2.2/2.4.1340-03, вопросы требований к защите от поражения электрическим током освещены в ГОСТ Р 12.1.019-2009 ССБТ.

Помещение, где расположено рабочее место оператора ПЭВМ, относится к помещениям без повышенной опасности ввиду отсутствия следующих факторов: сырость, токопроводящая пыль, токопроводящие полы, высокая температура, возможность одновременного прикосновения человека к имеющим соединение с землей металлоконструкциям зданий, технологическим аппаратам, механизмам и металлическим корпусам электрооборудования [15].

Основным организационным мероприятием по обеспечению безопасности является инструктаж и обучение безопасным методам труда, а также проверка знаний правил безопасности и инструкций в соответствии с занимаемой должностью применительно к выполняемой работе.

К мероприятиям по предотвращению возможности поражения электрическим током относятся:

- С целью защиты от поражения электрическим током, возникающим между корпусом приборов и инструментом при пробое сетевого напряжения на корпус, корпуса приборов и инструментов должны быть заземлены;
- При включенном сетевом напряжении работы на задней панели корпуса приборов должны быть запрещены;
- Все работы по устранению неисправностей должен производить квалифицированный персонал;
- Необходимо постоянно следить за исправностью электропроводки.

5.5 Экологическая безопасность

5.5.1 Влияние объекта исследования на окружающую среду

Так как объектом исследования в данной работе является информационная система, непосредственного влияния на экологию он не несет.

Тем не менее можно проследить опосредованное влияние на атмосферу, гидросферу и литосферу.

Использование ПЭВМ приводит к увеличению потребления электроэнергии, что влечет за собой увеличение мощностей электростанций, которые уже могут непосредственно загрязнять атмосферу, гидросферу и литосферу.

Также на гидросферу и литосферу оказывают негативное влияние отходы, образующиеся при производстве и утилизации ПЭВМ.

5.5.2 Мероприятия по обеспечению экологической безопасности

Для уменьшения вредного влияния на литосферу и гидросферу необходимо производить переработку отходов, возникающих при производстве и утилизации ПЭВМ.

В России на сегодня развито три направления утилизации электроники: рынок б/у электронных компонентов, вторичная переработка цветных металлов и вторичная переработка электроники, содержащей драгметаллы.

Также для снижения влияния на экологию необходимо стремиться к снижению энергопотребления, то есть разрабатывать и внедрять системы с малым энергопотреблением и минимизировать простои техники. Следует использовать современные ЭВМ с режимом пониженного потребления электроэнергии при длительном простое.

5.6 Безопасность в чрезвычайных ситуациях

Чрезвычайные ситуации, которые могут возникнуть при разработке и эксплуатации проектируемого решения:

- техногенные (взрывы, пожары, обрушение помещений, аварии);
- природные (наводнения, ураганы, бури, природные пожары);
- биологические (эпидемии, пандемии);
- антропогенные (война, терроризм).

Наиболее характерной чрезвычайной ситуацией для объекта, где размещаются офисные помещения, оборудованные электронно-вычислительными машинами, является пожар, так как в современных ЭВМ очень высокая плотность размещения элементов электронных схем. В непосредственной близости друг от друга располагаются соединительные провода и кабели, при протекании по ним электрического тока выделяется значительное количество теплоты, при этом возможно оплавление изоляции и возникновение возгорания.

При работе компьютерной техники выделяется много тепла, что может привести к пожароопасной ситуации. Источниками зажигания так же могут

служить приборы, применяемые для технического обслуживания, устройства электропитания, кондиционеры воздуха. Серьёзную опасность представляют различные электроизоляционные материалы, используемые для защиты от механических воздействий отдельных радиодеталей.

В связи с этим, участки, на которых используется компьютерная техника, по пожарной опасности относятся к категории пожароопасных «В».

При пожаре люди должны покинуть помещение в течение минимального времени.

В помещениях с компьютерной техникой, недопустимо применение воды и пены ввиду опасности повреждения или полного выхода из строя дорогостоящего электронного оборудования.

Для тушения пожаров необходимо применять углекислотные и порошковые огнетушители, которые обладают высокой скоростью тушения, большим временем действия, возможностью тушения электроустановок, высокой эффективностью борьбы с огнем. Воду разрешено применять только во вспомогательных помещениях.

5.7 Выводы по разделу

В ходе выполнения данного раздела были выявлены опасные и вредные факторы, которые могут возникнуть в процессе разработки и эксплуатации рассматриваемой информационной системы. Проведен анализ влияния рассматриваемого объекта на экологию и предложены меры по его минимизации. Также рассмотрены наиболее вероятные чрезвычайные ситуации и меры безопасности для ликвидации их последствий. Кроме того, рассмотрены правовые и организационные вопросы обеспечения безопасности при разработке информационно системы.

ЗАКЛЮЧЕНИЕ

В результате выполнения выпускной квалификационной работы был разработан серверный компонент и клиентское веб-приложение для информационной системы, позволяющей создавать эффект «оживающих» фотографий с использованием технологий дополненной реальности.

Серверный компонент был реализован в виде REST API. Разработка была выполнена на платформе NodeJS, с использованием веб-фреймворка Express.js. В качестве СУБД была использована PostgreSQL.

Клиентское веб-приложение было выполнено в виде одностраничного приложения, на основе библиотеки React. Разработанное приложение было адаптировано под мобильные устройства, при помощи отзывчивой верстки.

Разработанное веб-приложение позволяет клиентам создавать исходники (пары фото-видео), при помощи которых может быть осуществлен эффект «оживающих» фотографий.

Разработанный серверный компонент предоставляет доступ к исходникам мобильному приложению, реализующему технологию дополненной реальности.

CONCLUSION

As a result of the final qualification work, a server component and a client web application were developed for an information system that allows you to create the effect of “alive” photos using augmented reality technologies.

The server component was implemented as a REST API. The development was performed on the NodeJS platform using the Express.js web framework. PostgreSQL was used as the DBMS.

The client web application was created as a single page application based on the React library. The developed application was adapted for mobile devices using responsive web-design.

The developed web application allows customers to create source pairs (pairs of photo-video), with the help of which the effect of “alive” photos can be realized.

The developed server component provides access to the source pairs for a mobile application that implements augmented reality technology.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ГОСТ Р ИСО/МЭК ТО 10032-2007. Эталонная модель управления данными // Электронный фонд правовой и нормативно-технической документации URL: <http://docs.cntd.ru/document/gost-r-iso-mek-to-10032-2007> (дата обращения: 01.05.2020).
2. SQL или NoSQL — вот в чём вопрос // habr.com URL: <https://habr.com/ru/company/ruvds/blog/324936/> (дата обращения: 01.05.2020). ASP.NET [Электронный ресурс]. URL: <https://ru.wikipedia.org/wiki/ASP.NET> (дата обращения: 01.05.2020).
3. ASP.NET // Википедия свободная энциклопедия URL: <https://ru.wikipedia.org/wiki/ASP.NET> (дата обращения: 02.05.2020).
4. .NET is open-source on Github // <https://dotnet.microsoft.com/> URL: <https://dotnet.microsoft.com/platform/open-source> (дата обращения: 21.04.2020).
5. Рейтинг языков программирования 2020: JavaScript опередил Java // techrocks.ru URL: <https://techrocks.ru/2020/02/08/programming-languages-rank-2020/> (дата обращения: 01.05.2020).
6. Распространенные клиентские веб-технологии // docs.microsoft.com/ Техническая документация, материалы по API и примеры кода URL: <https://docs.microsoft.com/ru-ru/dotnet/architecture/modern-web-apps-azure/common-client-side-web-technologies> (дата обращения: 01.05.2020).
7. React // Википедия свободная энциклопедия URL: <https://ru.wikipedia.org/wiki/React> (дата обращения: 01.05.2020).
8. Трудовой кодекс Российской Федерации (с изменениями на 24 апреля 2020 года) // Электронный фонд правовой и нормативно-технической документации URL: <http://docs.cntd.ru/document/901807664> (дата обращения: 02.05.2020).
9. ГОСТ 22269-76 Система "Человек-машина". Рабочее место оператора. Взаимное расположение элементов рабочего места. Общие эргономические требования // Электронный фонд правовой и нормативно-

технической документации URL: <http://docs.cntd.ru/document/1200012834> (дата обращения: 02.05.2020).

10. ГОСТ 12.2.032-78 ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования // Электронный фонд правовой и нормативно-технической документации URL: <http://docs.cntd.ru/document/1200003913> (дата обращения: 02.05.2020).

11. Охрана труда при работе ЭВМ и видеодисплейных терминалов // works.doklad.ru Учебные материалы URL: <https://works.doklad.ru/view/2D8ku6NAGJM.html> (дата обращения: 02.05.2020).

12. Опасные и вредные производственные факторы // grandars.ru Энциклопедия Экономиста URL: <http://www.grandars.ru/shkola/bezopasnost-zhiznedeyatelnosti/opasnye-proizvodstvennyye-factory.html> (дата обращения: 02.05.2020).

13. СанПиН 2.2.4.548-96 Гигиенические требования к микроклимату производственных помещений // Электронный фонд правовой и нормативно-технической документации URL: <http://docs.cntd.ru/document/901704046> (дата обращения: 02.05.2020).

14. СанПиН 2.2.2/2.4.1340-03. Гигиенические требования к персональным электронно-вычислительным машинам и организации работы // Электронный фонд правовой и нормативно-технической документации URL: <http://docs.cntd.ru/document/901865498> (дата обращения: 02.05.2020).

15. ГОСТ 12.1.013-78 Система стандартов безопасности труда (ССБТ). Строительство. Электробезопасность. Общие требования // Электронный фонд правовой и нормативно-технической документации URL: <http://docs.cntd.ru/document/5200308> (дата обращения: 02.05.2020).