**Содержание**

Введение 4

1. Постановка задачи 5
   1. Описание предметной области 5
   2. Актуальность решаемой задачи 5
   3. Характеристика решаемой задачи 6
2. Проектирование программного продукта 7
   1. Разработка модели данных 7
   2. Выбор программного обеспечения 11
   3. Определение требований к техническим средствам 13
   4. Защита информации 14

3 Разработка программного обеспечения 15

3.1 Определение формы представления входных и выходных

данных 15

3.2 Разработка справочной системы 20

3.3 Тестирование программного модуля 22

3.4 Описание разрабатываемого программного продукта 26

4 Экономическая часть 21

5 Охрана труда 30

6 Энерго- и ресурсосбережение 40

Заключение 43

Список используемых источников 44

Приложение А (обязательное) Входные и выходные формы 45

Приложение Б (обязательное) Текст программы 50

Приложение В (обязательное) Результаты тестирования 81

Приложение Г (обязательное) Описание программы 85

Приложение Д (обязательное) Документация пользователя 86

Приложение Е (обязательное) Расчет затрат на оплату труда и

отчислений на социальные нужды

# **Введение**

Информационный сайт — это интернет-ресурс, на котором размещаются материалы информационного характера, ориентированные на читателя; источник информации, цель которого раскрыть конкретную тематику, проблематику или вопрос, донести информацию до аудитории, держать читателя в курсе новостей определенной тематики или тематик.

История занимает важное место в культуре и сознании людей, и изучение исторических событий не только оказывает влияние на понимание прошлого, но и помогает формировать представления о современности. Одним из ключевых моментов истории Беларуси в период Великой Отечественной войны является освобождение города Могилева.

Информационный сайт «Освобождение города Могилева» представляет собой проект, посвященный данному важному историческому событию. Целью данного проекта является собрание и систематизация информации об освобождении города Могилева, а также предоставление доступа к этой информации широкому кругу пользователей.

На страницах этого сайта будут представлены факты, документы, фотографии и другие материалы, связанные с освобождением Могилева от фашистской оккупации в ходе Великой Отечественной войны. Пользователи смогут ознакомиться с хронологией событий, узнать о героях и подвигах, связанных с этим периодом истории.

Создание данного информационного сайта позволит сохранить и передать будущим поколениям память о героической истории освобождения города Могилева, а также познакомить широкую аудиторию с этим важным этапом истории Беларуси.

**1 Постановка задачи**

**1.1 Описание предметной области**

Предметная область данного дипломного проекта тесно связана с историческим событием освобождения города Могилева от немецко-фашистских захватчиков в период Великой Отечественной войны, который имел огромное стратегическое значение в контексте военных действий на территории Беларуси. Создание информационного сайта по этой теме представляет собой важное достижение проекта, посвященного этому историческому периоду.

В рамках изучения данной предметной области будут рассмотрены различные аспекты освобождения Могилева. Будет проведен анализ исторического контекста и предыстории события, включая ключевые моменты, которые предшествовали освобождению города, а также период оккупации.

Особое внимание будет уделено хронологии событий освобождения Могилева, включая детальное описание боевых действий, участие советских войск и их союзников. Будут раскрыты имена и подвиги личностей и героев, игравших ключевую роль в освобождении города, включая военных командиров, бойцов и обычных жителей, совершивших подвиги в трудных условиях войны.

Также будет обсуждено влияние освобождения Могилева на дальнейшее развитие Великой Отечественной войны, реакция населения на это событие и его важность для истории Беларуси. Созданный информационный сайт станет надежным источником ценной и достоверной информации об освобождении города Могилева, предоставляя всем желающим возможность углубить свои знания об этом важном историческом моменте.

Дипломный проект по освобождению города Могилева от немецко-фашистских захватчиков в ходе Великой Отечественной войны представляет собой уникальную возможность исследовать и воссоздать этот значимый исторический момент. Развитие информационного сайта посвященного этому событию позволит сохранить и передать будущим поколениям важные факты и события, заложенные в истории города и страны в целом.

Важно также уделить внимание не только хронологии освобождения Могилева, но и рассмотреть последующие последствия этого события. Как изменилась жизнь города после освобождения, каков был вклад жителей в борьбу за свободу и независимость, и какие уроки можно извлечь из этой истории для настоящего и будущего.

Информационный сайт станет площадкой для широкого круга читателей, желающих глубже понять исторические события, связанные с Могилевом в годы Великой Отечественной войны. Анализ предметной области позволит раскрыть множество аспектов этого периода, подчеркнуть роль и значение каждого участника событий, и оценить влияние освобождения города на ход войны и дальнейшее развитие.

Таким образом, исследование предметной области освобождения Могилева в контексте Великой Отечественной войны позволит создать информационный ресурс, который будет не только информативным, но и отразит всю глубину и значимость этого исторического события для Беларуси и всего мира.

**1.2 Актуальность решаемой задачи**

Создание информационного сайта «Освобождение города Могилева» позволит:

- собрать и систематизировать исторические материалы: Создание сайта позволит централизованно хранить и предоставлять доступ к различным источникам, документам, фотографиям, связанным с освобождением Могилева от немецко-фашистских захватчиков. Это поможет исследователям, студентам, учащимся и просто интересующимся легче найти нужную информацию.

- привлечь внимание к историческому событию: Информационный сайт будет способствовать повышению осведомленности об историческом значимом событии освобождения Могилева и привлечет внимание широкой аудитории к этой теме. Это поможет сохранить и передать память о героическом подвиге советских войск и жителей города.

- обеспечить образовательную и патриотическую функции: Информационный сайт будет не только источником знаний о важном историческом периоде, но также сможет служить средством патриотического воспитания, помогая поддерживать интерес к истории своей страны у различных поколений и формировать уважение к прошлому.

- продвигать туризм и культурный обмен: Информационный сайт может помочь привлечь внимание туристов к историческим местам Могилева, связанным с освобождением города, способствовать развитию культурного туризма и обмена опытом между различными регионами.

Таким образом, создание информационного сайта «Освобождение города Могилева» имеет не только культурно-историческое значение, но и обладает потенциалом стимулировать интерес к истории, образованию и патриотизму.

Целями и задачами сайта являются:

Сохранение исторической памяти: важно сохранять и передавать информацию о подвиге советских войск и жителей Могилева, которые сражались за освобождение города от фашистских захватчиков, чтобы не забыть и не потерять исторические уроки и ценности.

Образовательная ценность: Создание информационного сайта об освобождении Могилева позволит обогатить образовательный контент о Великой Отечественной войне, позволит широкой аудитории узнать о важности этого события и его влиянии на ход истории.

Патриотическое воспитание: Представление исторически правдивой информации о героическом прошлом своей страны способствует формированию патриотизма у молодого поколения, воспитывает уважение к подвигам предков и культурное наследие.

Информационная доступность: Интернет-ресурс об освобождении Могилева позволит сделать информацию о данном историческом событии более доступной и понятной для широкой аудитории, включая школьников, студентов, историков, и просто интересующихся историей.

Таким образом, дипломный проект по созданию информационного сайта «Освобождение города Могилева» обладает актуальностью как с точки зрения сохранения исторической памяти, так и с педагогической и образовательной точек зрения.

Уникальность информационного сайта «Освобождение города Могилева» может проявляться через несколько аспектов:

- Эксклюзивный контент: Сайт может содержать уникальные материалы, фотографии, документы, интервью с участниками событий или их родственниками, которые не представлены на других ресурсах. Это подчеркнет его ценность как авторитетного источника информации об освобождении Могилева.

- оригинальный дизайн и функционал: Уникальность сайта также может проявиться через оригинальный дизайн, интерактивные элементы, удобную навигацию и функциональные возможности, которые будут привлекать и удерживать посетителей.

- обширный и многосторонний контент: Сайт может охватывать различные аспекты освобождения Могилева – от хронологии событий до аналитических материалов, от личных историй участников боев до описания последствий освобождения для города. Это позволит создать полноценную и многофункциональную платформу для изучения и просвещения.

- взаимодействие с пользователями: Уникальность сайта может проявиться через активное взаимодействие с посетителями – возможностью комментирования материалов, обратной связи, онлайн-консультаций, а также организацию интерактивных мероприятий и конкурсов.

Таким образом, уникальность информационного сайта «Освобождение города Могилева» будет заключаться в его способности представить историческую информацию в увлекательной, доступной и интерактивной форме, привлечь внимание аудитории и оставить незабываемые впечатления у посетителей.

**1.3 Характеристика решаемой задачи**

Цель разработки информационного сайта - предоставить пользователям удобный доступ к информации об освобождении города Могилева времен второй мировой войны. Для этого необходимо реализовать следующие функции:

1. Общая информационная часть сайта:

* страницы с разбивкой по разным периодам;
* галерея фотографий времен второй мировой войны.
* подписываться на новостную рассылку

2. Административная часть:

* редактировать информацию;
* создавать новые страницы;
* добавлять фотографии в галерею.

Для развертывания информационного сайта важны следующие общие требования к программному обеспечению:

* Сайт должен работать на операционных системах Windows, MacOS и Linux;
* Нужна авторизации для доступа к административной панели, для предотвращения утечки данных.

Общая информационная часть сайта:

- главная страница нас встречает историей и легендой об основании города;

- «Начало» - история начала великой отечественной войны;

- «Буйничское поле» - название говорит само за себя, о чем будет повествовать данная страница. Один из множества примеров ожесточенной обороны своих позиций советскими войсками;

- «Борьба против захватчиков» - подпольная борьба и подвиги других людей, лишь частично связанных с армии. Как люди сплотились против общего врага;

- «Освобождение Могилева» - возвращение территорий, итоги войны и дальнейшее развитие после окончания ужасающих происшествий;

- «Галерея» - памятные фото с тех событий.

**2 Проектирование программного продукта**

**2.1 Разработка модели данных**

Все действия и данные, производимые сайтом, можно показать с помощью диаграмм UML. С помощью диаграммы вариантов использования проектируемая система представляется в виде множества сущностей или актёров, взаимодействующих с системой с помощью вариантов использования. Диаграмма вариантов использования – диаграмма, описывающая, какой функционал разрабатываемой программной системы доступен каждой группе пользователей.

Диаграмма прецедентов для администратора представлены на рисунке 1.

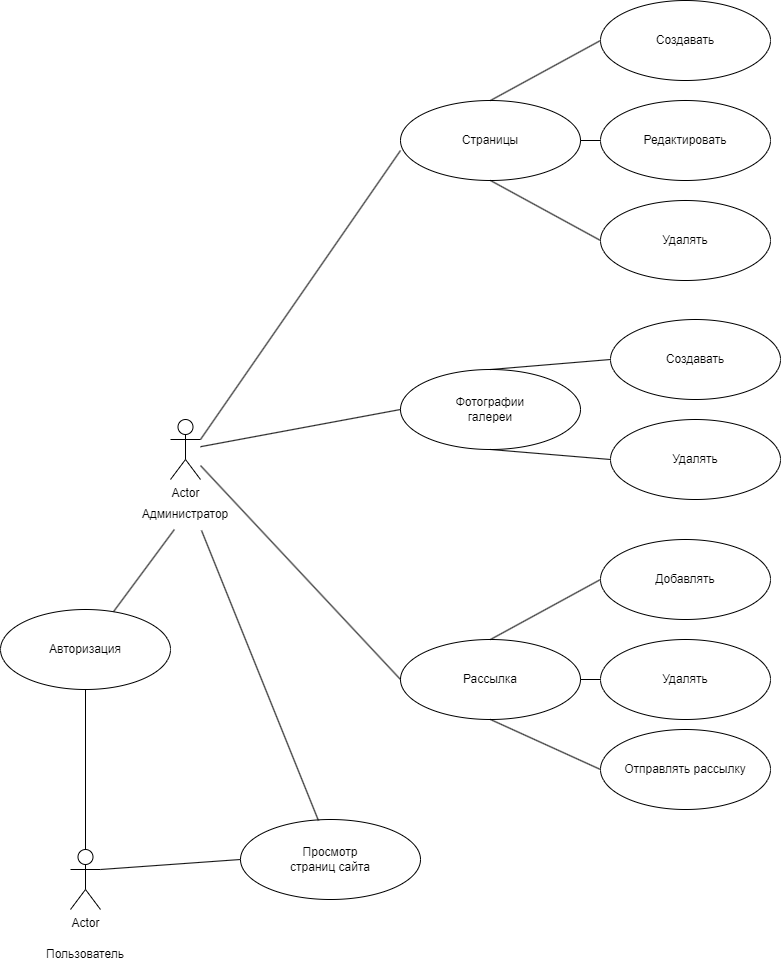


Рисунок 1 – Диаграмма прецедентов

Основным средством для предоставления статических моделей являются диаграммы классов на рисунке 2.

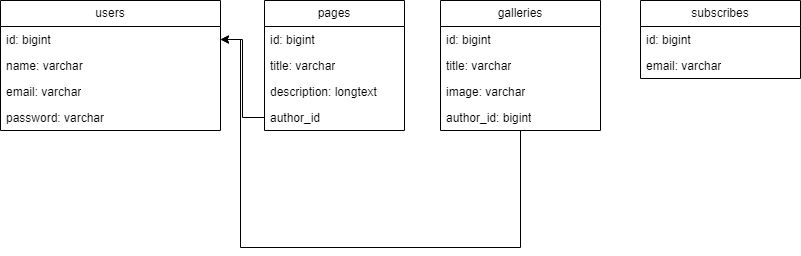


Рисунок 2 – Диаграмма классов

Статические модели обеспечивают представление структуры систем в терминах базовых строительных блоков и отношений между ними. «Статичность» этих моделей состоит в том, что здесь не показывается динамика изменений системы во времени. Вместе с тем, эти модели несут в себе не только структурные описания, но и описания операций, реализующих заданное поведение системы.

Последовательность действий для администратора представлены в виде UML-диаграмм на рисунках 3-5.

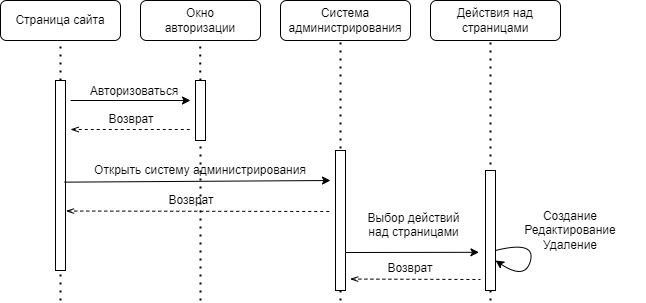


Рисунок 3 – Диаграмма последовательности для страниц

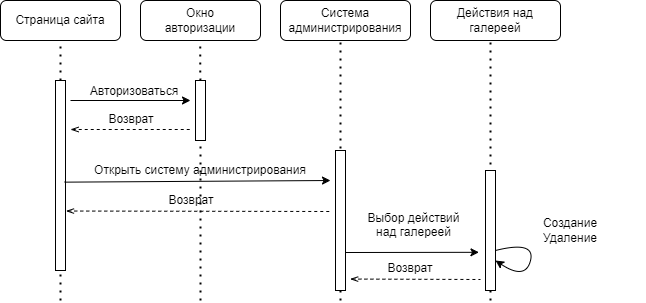


Рисунок 4 – Диаграмма последовательности для галереи

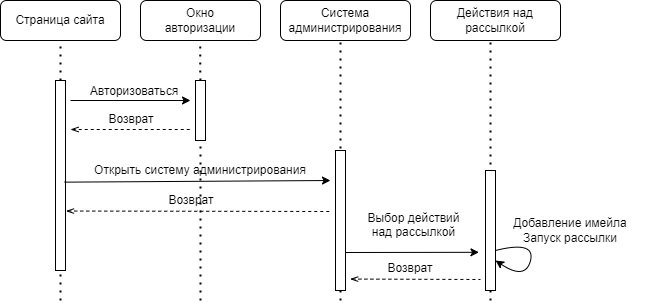
****

Рисунок 5 – Диаграмма последовательности для работы с рассылками

Диаграммы состояний для разных разделов сайт (общедоступной части сайта и административной панели) представлены на рисунках 6-7.

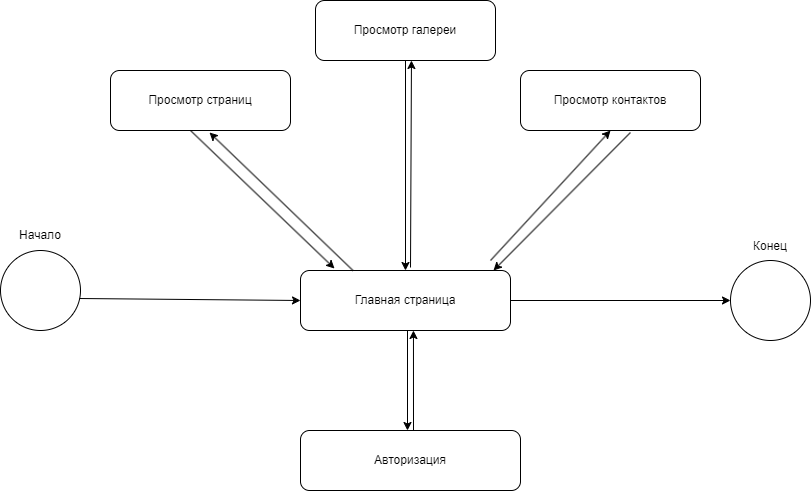


Рисунок 6 – Диаграмма состояний для общедоступной части сайта

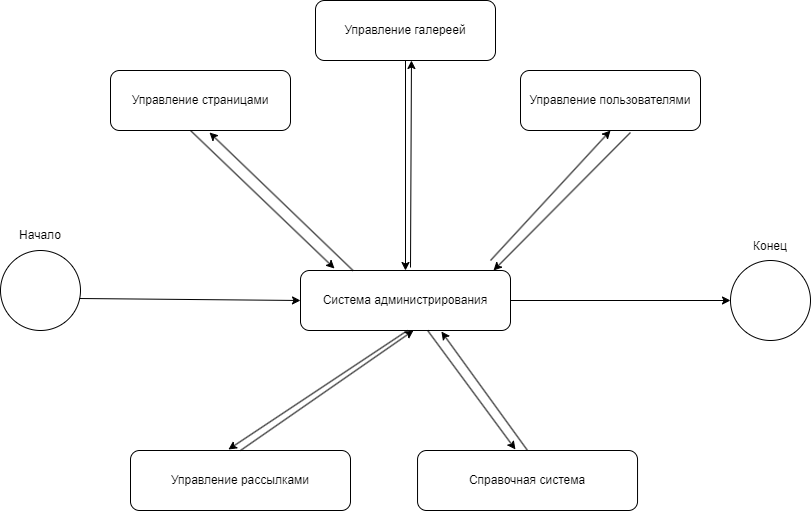
****

Рисунок 7 – Диаграмма состояний для административной части

Диаграммы развертывания и компонентов представлены на рисунках 8-9.

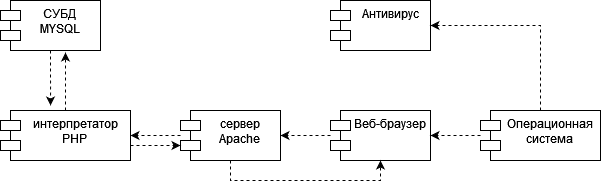


Рисунок 8 – Диаграмма компонентов

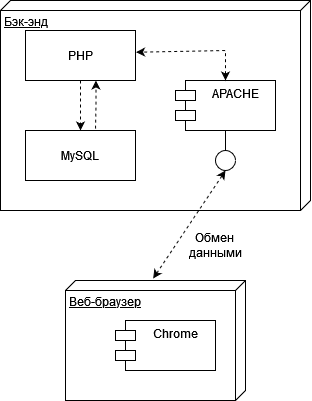


Рисунок 9 – Диаграмма развертывания

* 1. **Выбор программного обеспечения**

Информационный сайт разработан с помощью нескольких ключевых технологий и инструментов:

- Back-end с использованием Laravel:

Laravel — это фреймворк, который широко используется backend-разработчиками для создания веб-приложений на PHP. Он предоставляет структуру и каркас для разработки, основанный на архитектуре MVC (model-view-controller). Модель обеспечивает работу с данными, представление выводит данные пользователю, а контроллер управляет потоком данных в приложении.

- Front-end с использованием Tailwind CSS:

Tailwind CSS — это настраиваемый CSS-фреймворк, позволяющий создавать дизайн без необходимости переопределения стилей. Он предоставляет широкие возможности настройки с помощью утилит, адаптивных вариантов и инструментов для управления классами компонентов.

- СУБД MySQL:

MySQL — это свободная реляционная система управления базами данных, которая обеспечивает высокую скорость работы, простую систему безопасности и поддерживает различные типы таблиц. MySQL широко применяется при разработке веб-сайтов и приложений.

- Веб-сервер Apache:

Apache HTTP Server - бесплатное программное обеспечение для создания веб-серверов. Он используется для открытия динамических PHP-страниц, обеспечения стабильности сервера и распределения нагрузки. Apache является кроссплатформенным и поддерживает различные операционные системы.

Использование этих технологий позволяет создавать надежные веб-приложения, эффективно управлять данными и обеспечивать хороший пользовательский опыт.

**2.3 Определение требований к техническим средствам**

Минимальные требования для корректной работы OpenServer:

Требования к программному обеспечению:

* CPU Intel i3 2100;
* 1024 Мб ОЗУ;
* HDD 80Gb;
* Graphic card GeForce 5200Fx 128bit/128Mb;
* иметь доступ в интернет;
* операционная система любая;
* монитор 14;
* стандартная мышь;
* стандартная клавиатура;
* 1024 Мбайт свободного пространства на HDD;

Для безопасной работы программы и сохранности полученной информации рекомендуется использование источника бесперебойного питания.

**2.4 Защита информации**

Защита информации — это процесс обеспечения конфиденциальности, целостности и доступности данных.

Шифрование данных. Это процесс преобразования данных в зашифрованный вид, который может быть прочитан только с помощью ключа. Шифрование может использоваться для защиты конфиденциальных данных, таких как пароли, номера кредитных карт и другие личные данные.

Установка паролей. Это может быть пароль на доступ к компьютеру, пароль на доступ к файлам или пароль на доступ к сети. Пароли должны быть достаточно сложными и надежными для защиты от взлома.

Использование антивирусного программного обеспечения. Это поможет защитить компьютер от вирусов, троянов, шпионского ПО и других вредоносных программ.

Регулярное обновление программного обеспечения и операционной системы. Это поможет устранить уязвимости, которые могут быть использованы злоумышленниками для атаки на компьютер.

Резервное копирование данных. Это поможет сохранить данные в случае потери или повреждения жесткого диска или других носителей информации.

Используйте учетные записи, защищенные паролем: создайте учетные записи, защищенные паролем, для авторизованного персонала, который может получить доступ к результатам тестирования. Обязательно используйте надежные пароли и регулярно обновляйте их.

Шифрование: используйте шифрование для защиты данных. Это гарантирует, что даже если кто-то получит несанкционированный доступ к данным, он не сможет их прочитать.

Регулярные резервные копии: всегда сохраняйте резервные копии данных. Это гарантирует, что в случае нарушения безопасности вы сможете восстановить данные.

Внедрите меры контроля доступа. Внедрите меры контроля доступа, такие как брандмауэры, системы обнаружения вторжений и антивирусное программное обеспечение для защиты от несанкционированного доступа.

**3 Разработка информационного сайта**

**3.1 Определение формы представления входных и выходных данных**

При первом посещении сайта открывается главная страница сайта, представленная на рисунке 10.



Рисунок 10 – Главная страница сайта

Для того чтобы получить доступ к административной части нам необходимо авторизоваться по адресу «/login» или «/admin», во втором случае если пользователь ранее не был авторизован он будет автоматически перенаправлен на адрес «/login». Форма авторизации показана на рисунке 11.

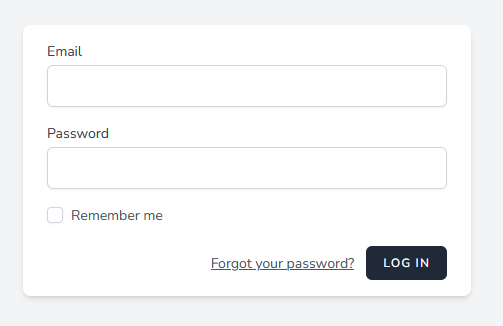


Рисунок 11 – Форма авторизации

После авторизации мы попадаем в административную часть сайта, показанную на рисунке 12.

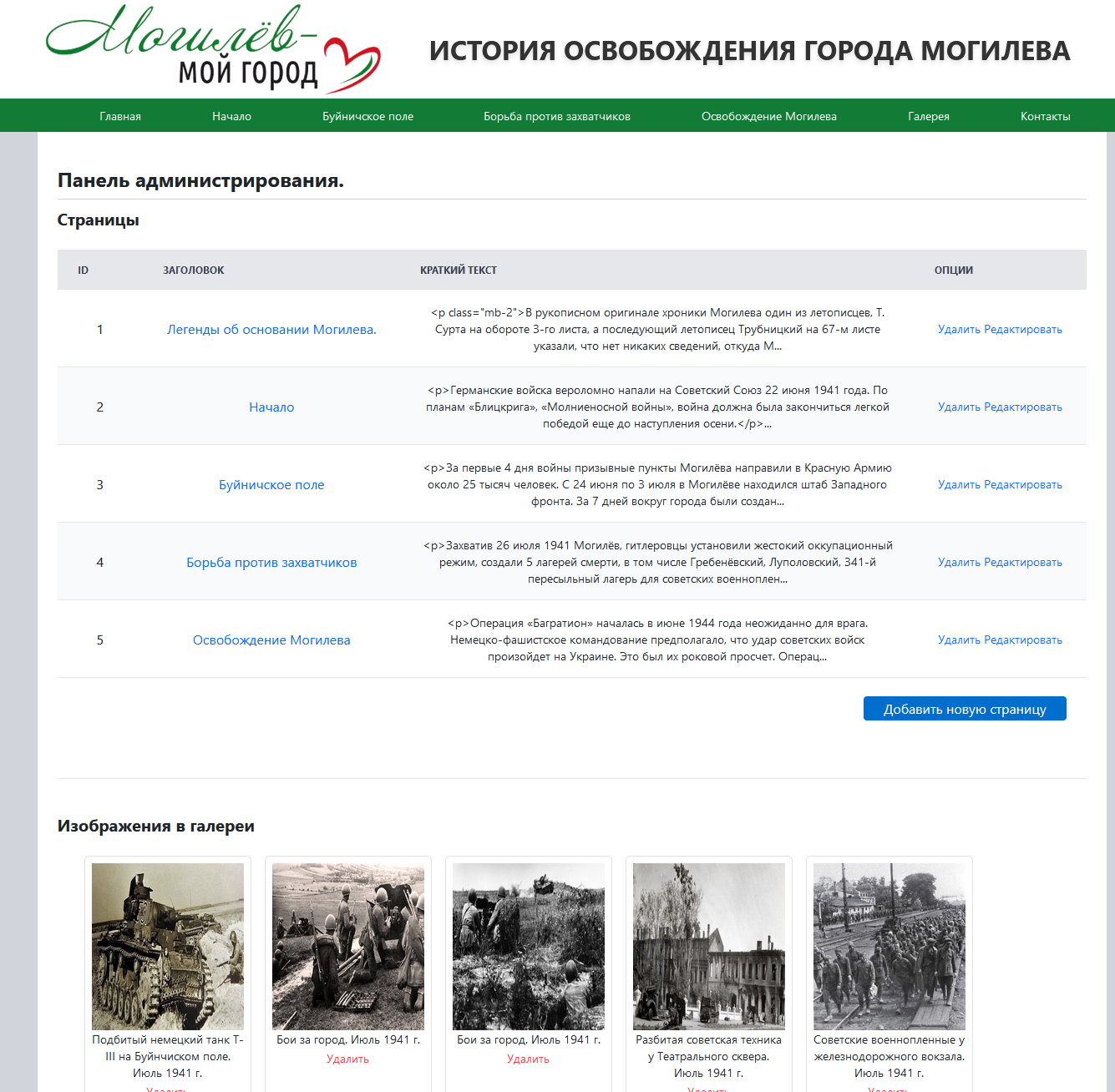


Рисунок 12 – Страница администрирования

Формы для добавления новых материалов на сайт представлены на рисунках 13-16.

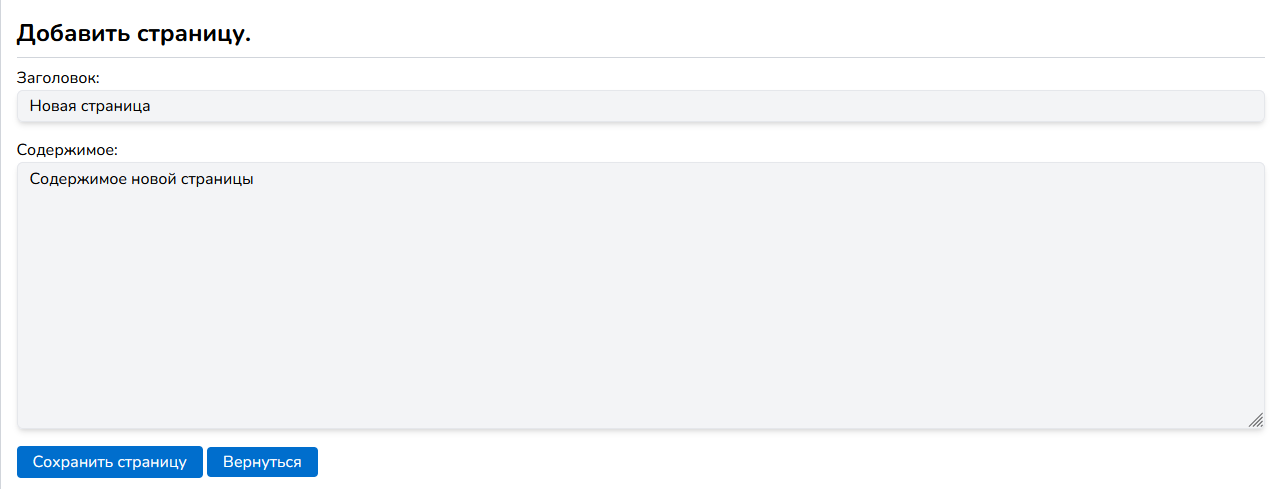


Рисунок 13 – Форма добавления страницы

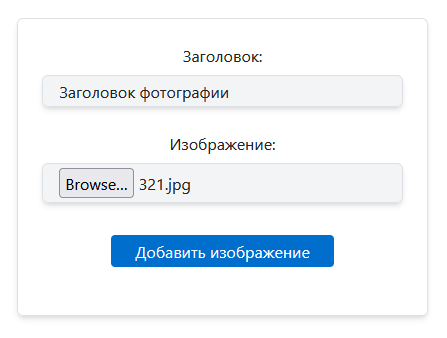


Рисунок 14 – Форма добавления фотографии в галерею

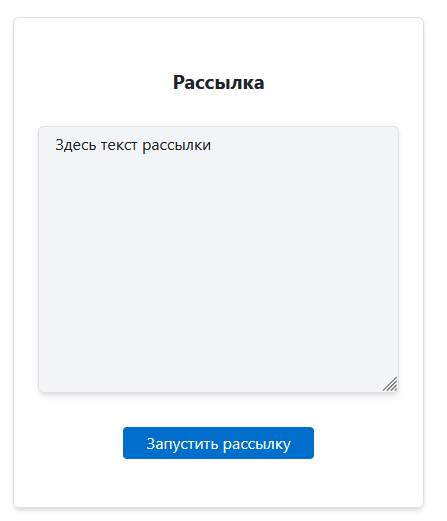


Рисунок 15 – Форма запуска рассылки

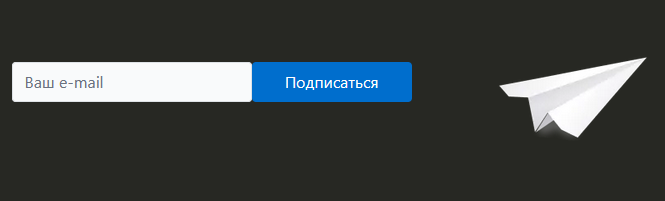


Рисунок 16 – форма подписки на рассылку

Передача всех форм данных осуществляется двумя из распространенных методов протокола HTTP и HTTPS – это GET-запрос и POST-запрос.

Запрос POST обычно отправляется через форму HTML и приводит к изменению на сервере.

В нашем случае POST запрос несет в себе информацию о добавление\редактировании сущности, такой как: информационная страница, фотография в галерею, добавление пользователя в список рассылки или выполнение самой рассылки. Так же формат POST-запроса используется для авторизации администратора на сайте и для разлогирования.

Для получения данных от сайта используется GET-запрос, при получении такого запроса происходит генерация HTML страницы с использованием бизнес-логики приложения, запросами к базе данных и отправки готовой HTML-страницы пользователю. С помощью GET-запроса можно только получать данные.

**3.2 Разработка справочной системы**

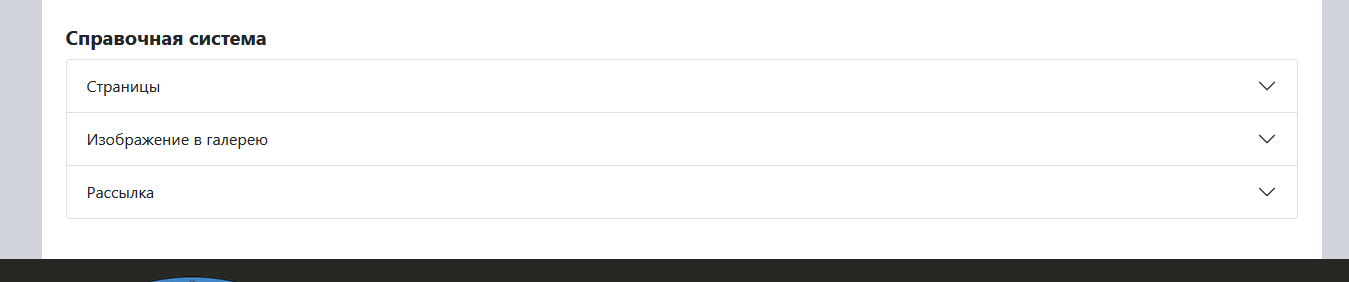
Для разработки справочной системы, мы интегрируем в админ-панель дополнительный раздел с названием «Справочная система». Формат справочный системы будет меню типа «аккордеон», где в каждом пункте будут подсказки о типах контента, пояснениями полей необходимых для создания данного типа контента.

Справочная система состоит из 3х пунктов:

- страницы;

- изображения в галереи;

- рассылка.



**Пункт «Страницы»**

В разделе страницы предоставляется возможность просмотра всех информационных страницы в табличном формате. В графе «Опции» представлены доступные опции для каждой страницы: редактировать и Удалить.

Редактирование организовано на отдельной странице, где в полях формы необходимо изменить нужные данные, которые хотите отредактировать после чего нажать кнопку «Сохранить». Удаление происходит сразу после нажатия кнопки «Удалить».

После таблицы доступных страниц находится кнопка «Добавить новую страницу», при нажатии на которую попадаешь на страницу добавление новой информационной страницы. Здесь необходимо заполнить 2 поля формы и нажать кнопку «сохранить страницу». После чего новая страница будет добавлена в базу данных и в меню появится новый пункт, ведущий на новую страницу

Пункт «Изображения в галереи»

В разделе «Изображения в галереи» предоставляется возможность просмотра всех изображений доступных для просмотра в галереи в формате «плитка». Сущность «Галерея» состоит из 3-х элементов:

- сама фотография;

- заголовок для фотографии;

- кнопка удалить.

При нажатии кнопку удалить происходит удаление фотографии из базы данных.

Ниже находится форма для добавления нового элемента для галереи. Заполняем поле «заголовок», выбираем изображение для загрузки и нажимаем кнопку «Добавить изображение»

Пункт «Рассылка»

В разделе «Подписки на рассылку» находится список пользовательских электронных адресов, которые подписались на рассылку. Данные предоставлены в табличном формате. В графе «Действия» находится кнопка удалить, при нажатии на которую происходит удаление пользовательского электронного адреса из базы данных

Ниже находится форма для рассылки сообщений всем пользователям, которые подписались на рассылку. Заполняем текстовую область сообщением, которое мы хотим разослать и нажимаем кнопку «Запустить рассылку»

**3.3 Тестирование программного модуля**

Тестирование программного обеспечения является важным процессом, требующим глубокого понимания особенностей программного продукта и требований к нему. Тестирование сайта состоит из 2-х этапов: функциональное тестирование, которое проверяет работы бэк-энда и «ручное», когда пользователь самостоятельно выполняет тестовые действия на сайте

Ручное тестирование

Основные пункты работы сайта которые необходимо проверить это:

1. работоспосбность сайта в целом;
2. проверка работоспособности формы авторизации;
3. проверка создания контента;
4. проверка выхода из системы;
5. проверка рассылки.

Результаты тестирования представлены в таблице 1.

Таблица 1 – Журнал тестирования

|  |  |  |
| --- | --- | --- |
| Действие актера | Действие бэк-енда | Отметка о правильной работе или описание ошибки |
| Открыть главную страницу | Запуск главного модуля приложения | Действие выполнено успешно |
| Перейти на страницу авторизации | Проверить валидность данных и проверить соответствие в базе данных введенных данных | Действие выполнено успешно |
| Создать новую страницу | Получить POST данные и внести изменения в базу данных | Действие выполнено успешно |
| Добавить изображение в галерею | Получить POST данные и внести изменения в базу данных | Действие выполнено успешно |
| Добавить пользовательский имейл в список рассылки | Получить POST данные и внести изменения в базу данных | Действие выполнено успешно |
| Создать пользователя | Получить POST данные и внести изменения в базу данных | Действие выполнено успешно |
| Выбрать пункт в админ-меню «выход» | Провести очистку данных в таблице sessions | Действие выполнено успешно |
| В админ-панели запустить рассылку | В цикле перебрать все электронные адреса из БД и отправить на них письмо | Действие выполнено успешно |

Функциональное тестирование

Функциональное тестирование — процесс обеспечения качества в рамках [цикла разработки программного обеспечения](https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D0%B0%D0%B4%D0%B8%D0%B8_%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B3%D0%BE_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D1%8F), необходимый для проверки реализуемости [функциональных](https://ru.wikipedia.org/wiki/%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D1%81%D0%BF%D0%B5%D1%86%D0%B8%D1%84%D0%B8%D0%BA%D0%B0%D1%86%D0%B8%D1%8F) [требований,](https://ru.wikipedia.org/wiki/%D0%A2%D1%80%D0%B5%D0%B1%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F_%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%BC%D1%83_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D1%8E) согласно [спецификации](https://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B5%D1%86%D0%B8%D1%84%D0%B8%D0%BA%D0%B0%D1%86%D0%B8%D1%8F) тестируемого [программного обеспечения](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5). Функциональное тестирование проводится для [оценки соответствия](https://ru.wikipedia.org/wiki/%D0%9E%D1%86%D0%B5%D0%BD%D0%BA%D0%B0_%D1%81%D0%BE%D0%BE%D1%82%D0%B2%D0%B5%D1%82%D1%81%D1%82%D0%B2%D0%B8%D1%8F) [системы](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0) или [компонента](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D0%BE%D0%BD%D0%B5%D0%BD%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) заданным функциональным требованиям.

Функциональное тестирование проводится по принципу [черного ящика](https://ru.wikipedia.org/wiki/%D0%A7%D1%91%D1%80%D0%BD%D1%8B%D0%B9_%D1%8F%D1%89%D0%B8%D0%BA), в связи с чем функциональность ПО можно протестировать, не зная принципа его внутренней работы. Это снижает требования к [тестировщикам](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D1%81%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D1%89%D0%B8%D0%BA) в части знания [языков программирования](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F) или конкретных аспектов реализации [программного обеспечения](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5).

В Laravel предусмотрены 3 типа тестов: unit, feature и browser тесты. Для нашей задачи были выбраны Feature тесты — это тесты, которые фокусируются на очень небольшой изолированной части нашего кода. Тесты должны покрывать 4 сущности нашего сайта:

Страницы:

- просмотр страниц сайта;

- создание страницы сайта;

- редактирование существующей страницы;

- удаление страницы.

Изображения в галереи:

- просмотр галереи;

- удаление изображения;

- добавление нового изображения;

Пользователи:

- создание нового пользователя;

- редактирование существующего пользователя;

- удаление пользователя;

- авторизация пользователя;

Подписки на рассылку:

- добавление нового электронного адреса в список;

- создание рассылки;

- удаление электронного адреса из списка.

Все тесты разбиты на 6 файлов кода, в которых находятся 15 методов для проверки работоспособности нашего проекта: отправка запроса с данными и получение ответа, если получен ответ от сервера «200» значит тест пройден успешно, либо правильным ответам может быть статус «301» - редирект на нужную страницу после выполнения действий

Так же для корректной работы тестирование мы создаем отдельный файл переменных окружения «.env\_testing» и отдельную базу данных, чтобы тестирование во время проверок не поломало основную базу данных и все данные остались в целости.

Результаты тестирования отображены на рисунке 17:

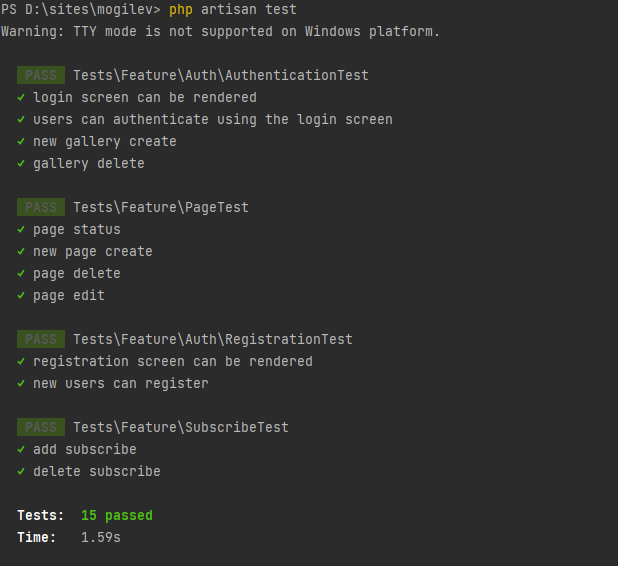


Рисунок 17 – Результаты тестирования

Все 15 тестов пройдены успешно, за время 1.59с. Время тестирования может варьироваться в зависимости от конфигурации серверного оборудования. Для данных результатов тестов использовалась следующая конфигурация оборудование:

CPU: Intel Xeon E5-2699 v3

Memory: 64Gb DDR4 ECC, quad channel mod

SSD m2 512Gb

Motherboard: Attermiter x99

GPU: NVidia RTX 2060 Super 8Gb  
Server: Apache 2.4

PHP: 8.1

ОС: Windows 10

Весь код тестовых модуль представлен в Приложении В.

**3.4 Описание разрабатываемого программного продукта**

На главной странице представленной на рисунке 18 находится информация об истории города Могилева.



Рисунок 18 – Главная страница

Информация для этой странице находится в базе данных в таблице ‘pages’ с индексом 1. Все остальные страницы по шаблону “/page/ID” загружают данные из базы данных по индексу ID, которые содержат данные состоящие из 2 полей «Заголовок» и «Содержимое».

Страница «Галерея» содержит множество фотографий которые представлены миниатюрами в ряд – рисунок 19. При нажатии на фотографию срабатывает JavaScript для увеличения изображения для детального просмотра. Для этого использована библиотека JQuery и плагин LightBox.

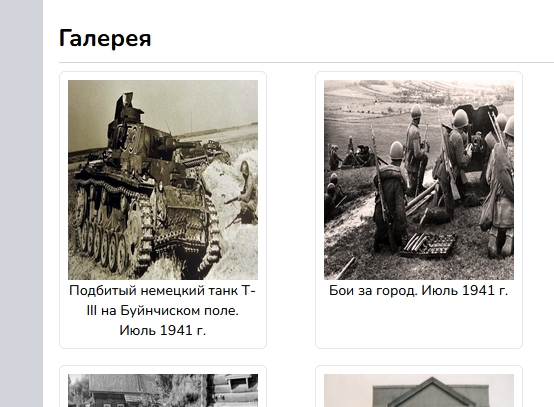


Рисунок 19 – Страница «Галерея»

Страница «Контакты» содержит интерактивную карту города Могилева – рисунок 20, которая загружается из сервиса Яндекс.Карты.

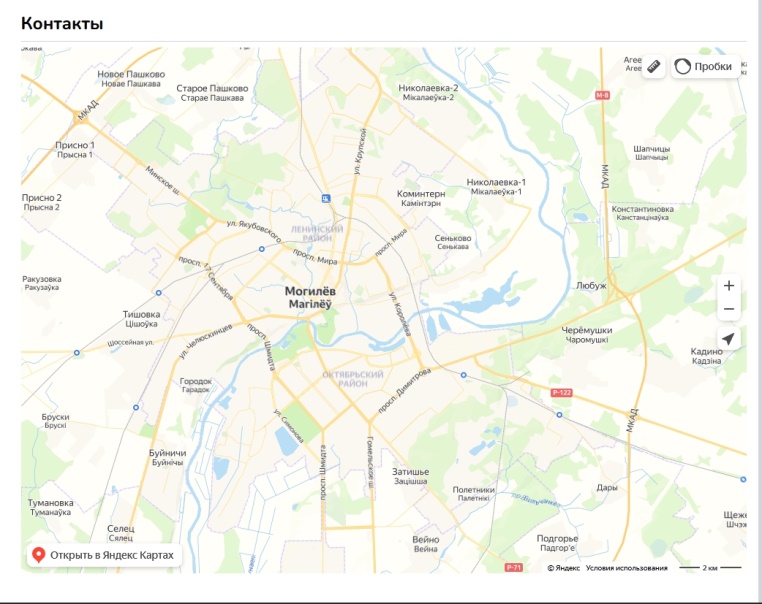


Рисунок 20 – Страница «Контакты»

Навигационное меню в шапке, сайдбаре и подвале генерируется автоматически в зависимости от страниц, которые созданы администратором. При добавлении новых страниц пункт в меню добавляется автоматически.

В подвале находится форма подписки для рассылки на новости показанная на рисунке 21. При вводе вашего email-адреса он заносится в базу данных и в дальнейшем будет участвовать в получении писем при отправке в административной панели.

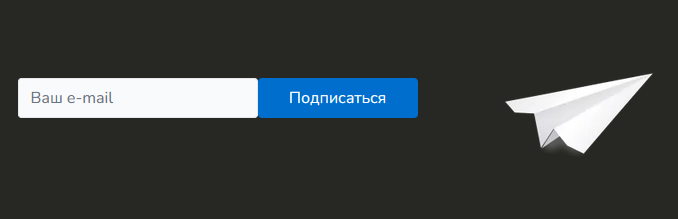


Рисунок 21 – Форма подписки на новости

Административная панель сайта

Административная панель доступна по адресу /admin, для доступа к которой необходимо пройти процедуру авторизации, окно авторизации представлено на рисунке 22.

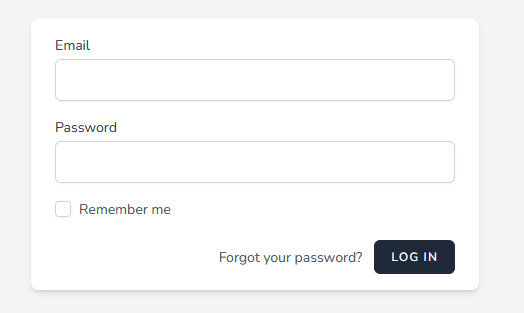


Рисунок 22 – Форма авторизации

После успешной авторизации мы попадаем на страницу администрирования которая делится на 5 разделов:

1. Страницы
2. Галерея
3. Пользователи
4. Подписка на рассылку
5. Справочная система

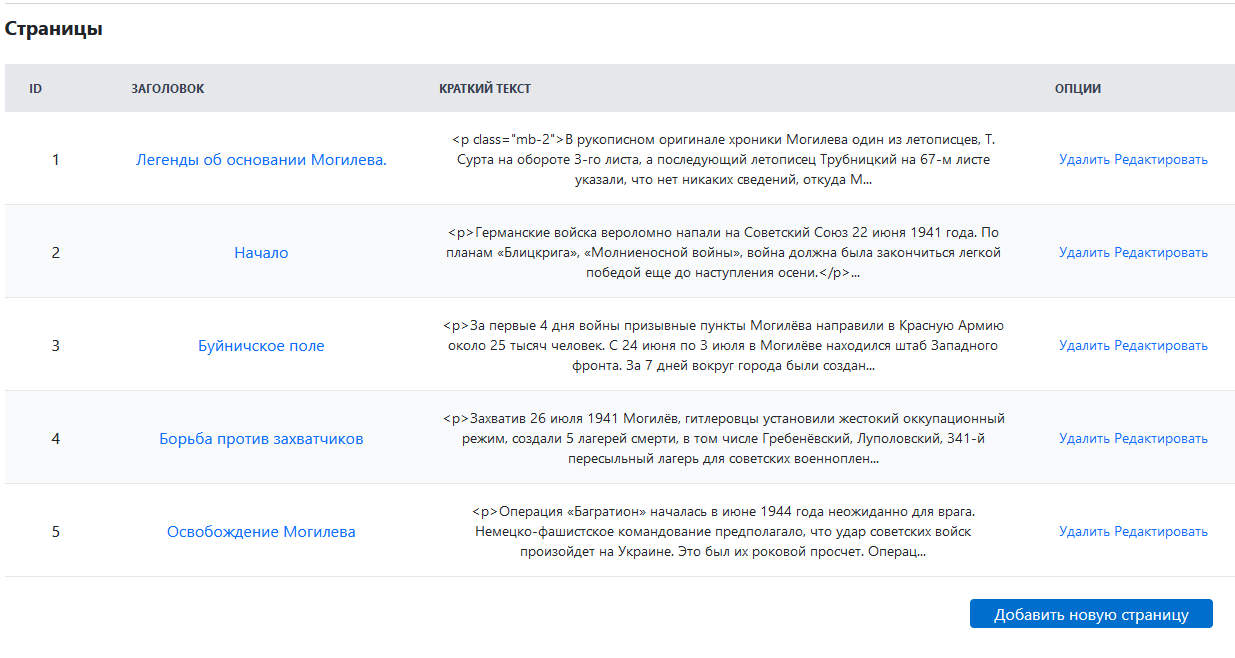


Рисунок 23 - Раздел «Страницы»

В этом разделе представлены в табличной форме все созданные администратором страницы. Поле «заголовок» является анкором для перехода на выбранную страницу. В поле «опции» выбирается нужное действие для этой страницы, доступные действие: удалить и редактировать. При нажатии на опцию «редактировать» администратор попадает на отдельную страницу редактирования данной страницы – рисунок 24.

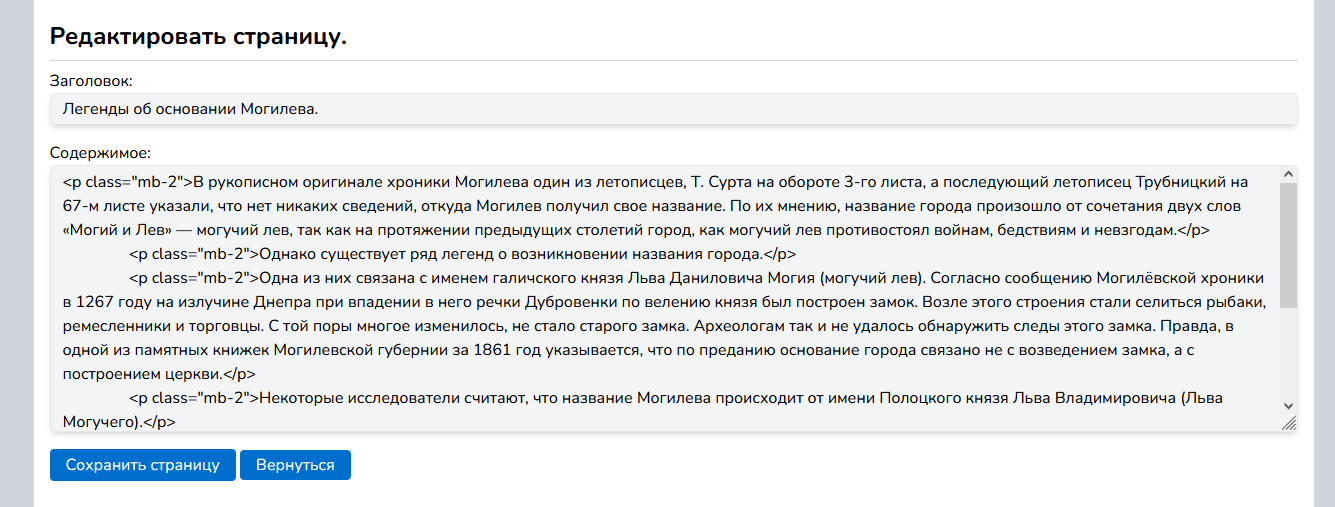


Рисунок 24 – Раздел редактирования страницы

После внесения всех изменений необходимо нажать кнопку «Сохранить страницу» для внесения изменений в базу данных.

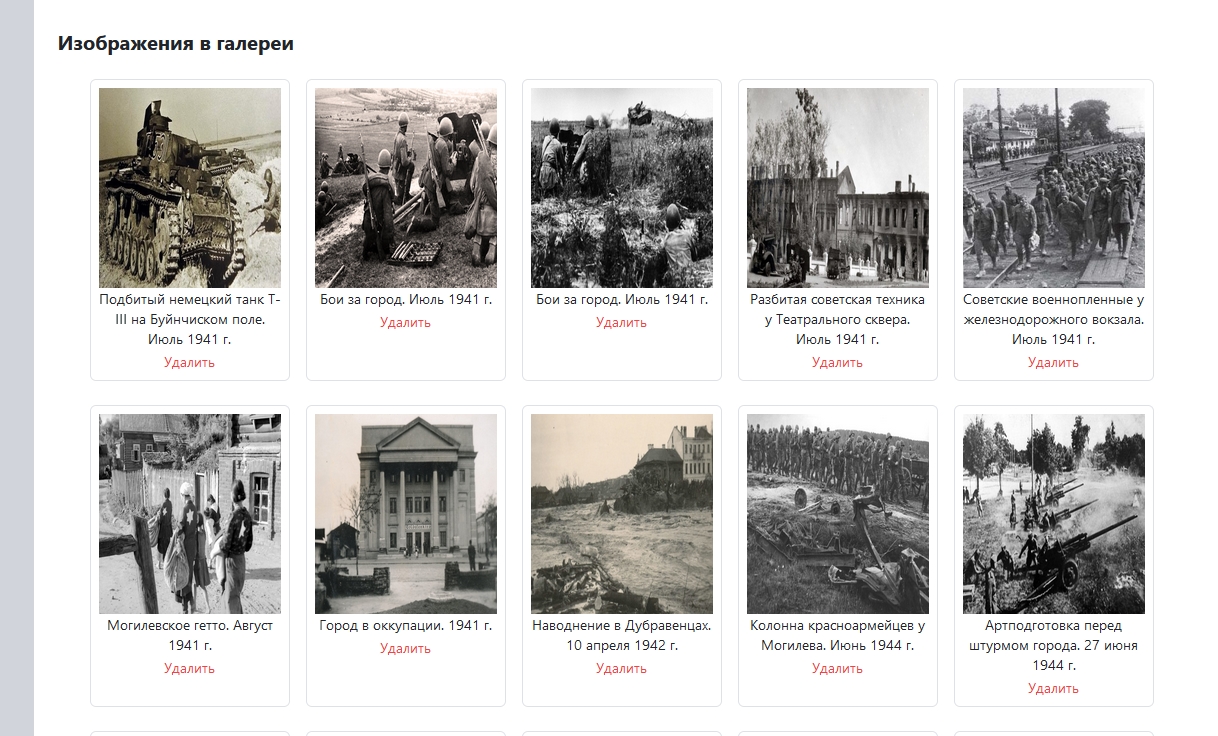
В начале раздела представлены все фотографии с подписями, которые отображаются на страницы сайта «Галерея». Из опций для данного типа материала предусмотрена кнопка «удалить», при нажатии на которую происходит удаление изображение из базу данных, а также физически из директории сайта – рисунок 25. 

Рисунок 25 – раздел редактирования галереи

На рисунке 26 располагается форма для добавления нового изображения.

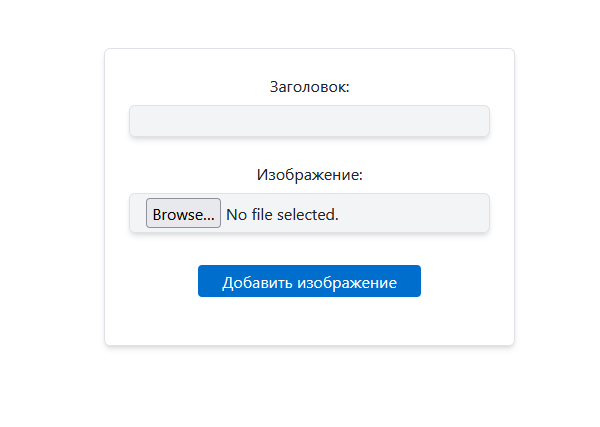


Рисунок 26 – форма добавления изображения

В поле заголовок мы должны указать название изображения, которое будет показываться на странице галереи и в развернутом состоянии при использовании Lightbox. В поле изображение необходимо выбрать файл картинки которое необходимо загрузить. После заполнения всех полей нажимаем кнопку «Добавить изображение», далее происходит загрузка файла на сервер и запись в базу данных.

Раздел «Пользователи»

В разделе «Пользователи», рисунок 27, представлены все пользователи, зарегистрированные на сайте.

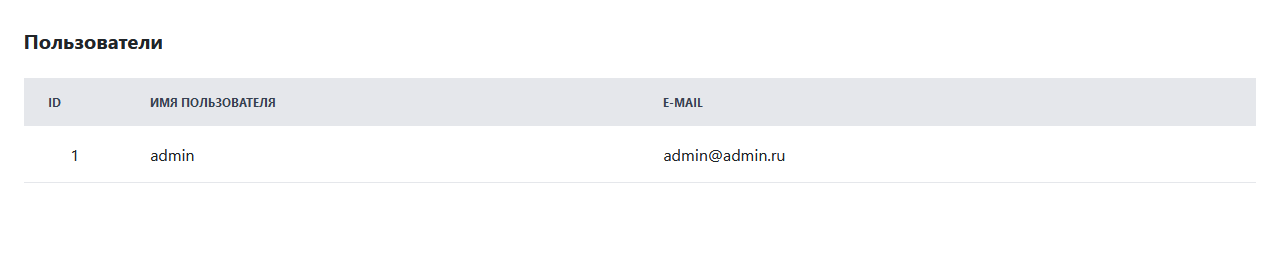


Рисунок 27 – Раздел «Пользователи»

В начале раздела «Подписка на рассылку», рисунок 28, отображена таблица всех пользователей, которые ввели свой электронный адрес и подписались на новостную рассылку, после чего расположена форма для отправки сообщений этому списку пользователей

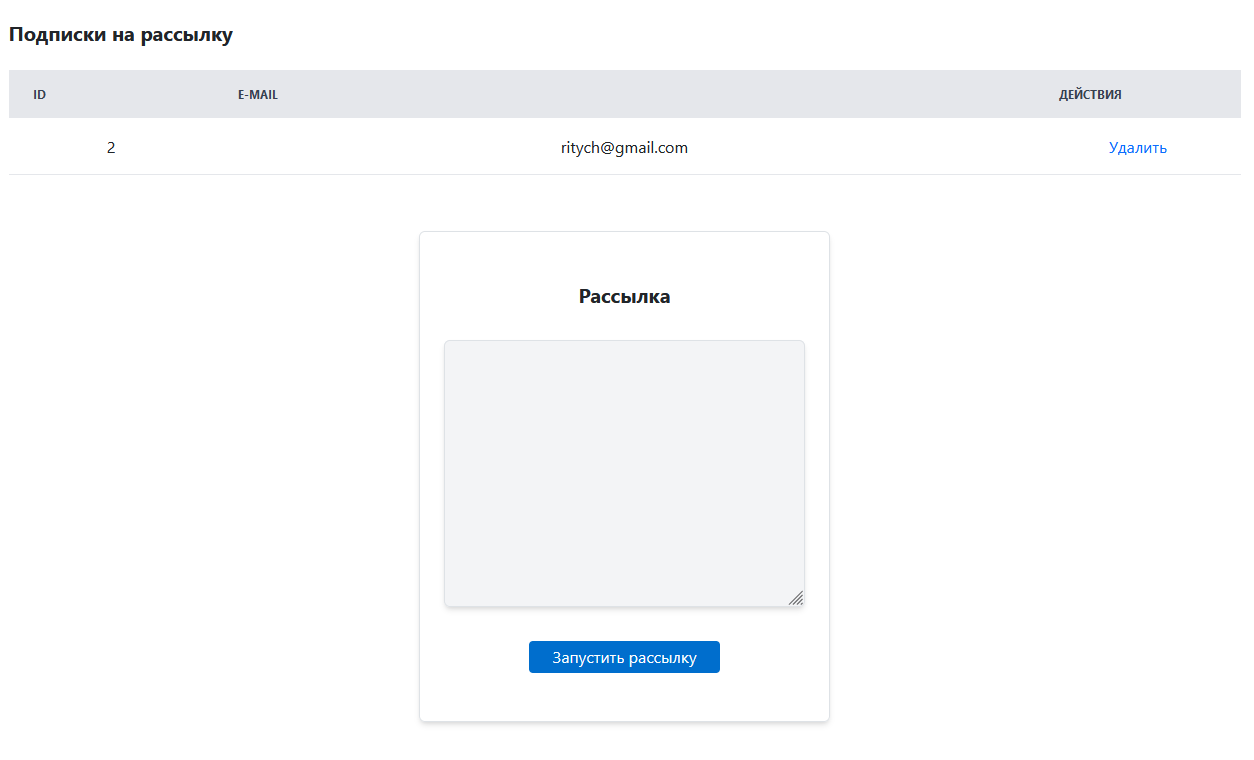


Рисунок 28 – Раздел «Подписка на рассылку»

Необходимо написать сообщение и нажать кнопку «Запустить рассылку», после чего это сообщение будет разослано по всем подписанным адресам.

Последовательность и расчет трудоемкости разработки программного продукта оформляется в виде таблицы 2.

Таблица 2-Трудоемкость разработки программного продукта

|  |  |  |  |
| --- | --- | --- | --- |
| Виды работ | Кол-во  операций | Норма времени, ч | |
| на одну  операцию | на все  операции |
| 1 | 2 | 3 | 4 |
| 1. Подготовка исходных данных | 14 | 0,3 | 4,2 |
| 1. Реализация алгоритмов контрольных задач с использованием ПС ПЭВМ | 12 | 0,3 | 3,6 |
| 1. Обработка данных и получение результатов | 9 | 0,28 | 2,52 |
| 1. Анализ ошибок обработки данных и подготовка заключения о результатах проверки | 8 | 0,35 | 2,8 |
| 1. Ознакомление сотрудников службы сопровождения с содержанием задач, структурой входных и выходных данных | 5 | 0,79 | 3,95 |
| 1. Определение параметров настройки | 5 | 0,17 | 0,85 |
| 1. Ознакомление с объектом внедрения | 1 | 0,94 | 0,94 |
| 1. Консультации по подготовке пользователями исходных данных в соответствии с требованиями и ограничениями ОС ПЭВМ | 7 | 1 | 7 |
| 1. Разработка рекомендаций по реализации алгоритмов и требований пользователя к обработке данных с использованием ППП ПЭВМ по подготовке задач к опытной эксплуатации | 6 | 2,2 | 13,2 |
| 1. Оценка соответствия функциональных и эксплуатационных характеристик ПС требованиям к обработке данных | 5 | 1,1 | 5,5 |
| 1. Проведение консультаций и анализ ошибок комплексирования в период опытной эксплуатации | 2 | 1,7 | 3,4 |
| 1. Разработка рекомендаций по созданию программных средств сопряжения (программ и блоков пользователя, осуществляющих промежуточную обработку данных) | 3 | 1,7 | 5,1 |
| 1. Корректировка программ с целью изменения незначительных функциональных характеристик | 6 | 1,06 | 6,36 |
| 1. Разработка дополнительных модулей и включение их в состав ПС | 2 | 2,4 | 4,8 |
| 1. Анализ требований задач пользователя к обработке данных и характеристик среды их функционирования | 4 | 0,7 | 2,8 |
| 1. Разработка требований к тестированию и подготовка тестовых единиц | 6 | 0,7 | 4,2 |
| 1. Анализ результатов прогона и разработка функциональных спецификаций на корректировку ПС | 5 | 0,6 | 3 |
| 1. Внесение изменений в программы и эксплуатационную документацию ПС у пользователя | 4 | 0,29 | 1,16 |
| 1. Демонстрация функционирования на контрольных задачах службы сопровождения | 5 | 0,2 | 1 |
| 1. Анализ организационно- экономических и технических характеристик объекта внедрения ПС | 1 | 0,4 | 0,4 |
| 1. Разработка требований к выбору ПС для реализации задач пользователя | 2 | 0,4 | 0,8 |
| 1. Рекомендации по выбору ПС | 1 | 0,23 | 0,23 |
| 1. Оценка полноты охвата функциональными возможностями ПС | 3 | 0,4 | 1,2 |
| 1. Требования задач пользователей данного класса | 4 | 0,4 | 1,6 |
| 1. Выработка рекомендаций по расширению функциональных возможностей ПС | 3 | 0,1 | 0,3 |
| 1. Оценка необходимости проведения обучения работе по утвержденной технологии | 4 | 0,5 | 2 |
| 1. Анализ характеристик и производственных условий разработки, изготовления и сопровождения ПС у пользователя | 3 | 0,6 | 1,8 |
| 1. Разработка рекомендаций по применению у пользователя поставляемых технологических процессов разработки, производства и сопровождения ПС | 2 | 1,1 | 2,2 |
| 1. Практическая работа пользователей с ПС в вычислительном центре службы сопровождения | 6 | 1,4 | 8,4 |
| 1. Разработка рекомендаций по подготовке исходных данных в соответствии с требованиями и ограничениями ПС, по созданию технологии обработки данных с помощью ПС | 3 | 0,58 | 1,74 |
| 1. Консультации по устранению ошибок подготовки данных и решению задач | 5 | 0,3 | 1,5 |
| 1. Вывод на печать | 97 | 0,0028 | 0,27 |
| Итого трудоемкость  в т.ч ПЭВМ,  принтер |  |  | 98,82  98,55  0,27 |

Расход бумаги составит 97 листов, носителей информации – 1 диск DVD‑R.

Кроме программного модуля разработана сопровождающая программная документация в соответствии с ГОСТ 19.402-2000 «Описание программы», которая представлена в приложении Г и документация пользователя в соответствии с ГОСТ ИСО 9127-2002 «Документация пользователя», которая представлена в приложении Д.

**4 Экономическая часть**

**4.1 Расчет материальных затрат**

К материальным затратам относятся затраты на расходные материалы и затраты на электроэнергию на технологические цели.

Материальные затраты МЗ, руб., рассчитываются по формуле

МЗ = Ср.м + Сэн,

где Ср.м - стоимость расходных материалов, руб.;

Сэн - стоимость электроэнергии, руб.

МЗ = 35,95 + 15,41 = 51,36 руб.

Затраты на расходные материалы Ср.м, руб., определяются по формуле

Ср.м = Сб + Ск + Сн,

где Сб - стоимость бумаги, руб.;

Ск - стоимость картриджа для принтера, руб.;

Сн - стоимость носителя информации, руб.

Ср.м = 3,30 + 30,50 + 2,15 = 35,95 руб.

Затраты на бумагу определяются по формуле

Сб = Цб × Рб,

где Цб - цена за 1 лист бумаги, руб,;  
 Рб - расход листов бумаги при разработке и печати программного

продукта, шт;

Сб = ×97= 3,30 руб.

Затраты на носители информации определяются по формуле

Сн = Цн × Рн,

где Цн - цена носителя информации, руб.;

Рн - расход дискет, CD, CD-RW, шт.

Сн =2,15×1 = 2,15 руб.

Затраты на электроэнергию определяют исходя из загруженности персонального компьютера программиста и частично занятости принтера за время разработки (учитывая, что при разработке программного продукта принтер используется меньше, чем персональный компьютер).

Затраты на электроэнергию определяются по формуле

Сэн = Цэн × (Тпк × Wпк + Тприн × Wприн),

где Цэн - тариф за 1 кВт-ч электроэнергии, руб.;

Тпк - время работы персонального компьютера, ч;

Тприн - время работы принтера, ч;

Wпк - потребляемая мощность ПК, кВт-ч;

Wприн - потребляемая мощность принтера, кВт-ч.

Сэн = 0,39×(98,55×0,40+0,27×0,35) = 15,41 руб.

**Расчет затрат на оплату труда и отчислений на социальные нужды**

Расчет затрат на оплату труда и отчислений на социальные нужды осуществляется при помощи прикладной программы на ПЭВМ. Пример данного расчета приведен в приложении Е.

**Расчет себестоимости разработки программного продукта**

Себестоимость разработки программного продукта Спол., руб., рассчитывается по формуле

Спол=МЗ+ФОТ+Осн+Ао+Зпр,

где Ао – амортизационные отчисления основных средств и

нематериальных активов,руб .;

Зпр – прочие затраты ,руб.

Спол = 51,36+ 785,02+266,91+6,75+2424,84=3534,90 руб.

Амортизационные отчисления рассчитываются по формуле

Ао = ,

К = ,

Фд = Фн×(1-%)

где ОС – стоимость основных средств и нематериальных активов , 1350,00 руб.;

Тн – нормативный срок службы ,10 лет ;

К – коэффициент, учитывающий долю занятости ПЭВМ;

Фд – действительный фонд времени работы ПЭВМ, ч;

Фн – утвержденный номинальный годовой фонд времени ,2016 ч;

%П – процент простоя оборудования в ремонте ,%;

Фд = 2016×(1- ) = 1874,88 ч.

К = = 0,05

АО = = 6,75 руб

Прочие затраты включают оплату услуг связи ,ВЦ, банков, сигнализаций , консультаций, аудиторских и рекламных услуг , за охрану ; вознаграждение ра-ботникам за изобретения и рационализаторские предложения; арендная плата за имущество; плата по процентам за краткосрочный и долгосрочный кредиты под пополнение оборотных средств; земельный налог; налог за пользование природ- ными ресурсами и другие налоги ,включаемые в себестоимость , рассчитываются по формуле

Зпр = ,

где %Зпр – процент прочих затрат ,%;

Зпр =  = 2431,45 руб

Результаты расчетов заносим в таблицу 3.

Таблица 3. Результаты расчетов

|  |  |
| --- | --- |
| Наименование  элементов затрат | Сумма, руб. |
| 1 Материальные затраты (за вычетом стоимости возвратных отходов) | 51,36 |
| 2 Затраты на оплату труда | 785,02 |
| 3 Отчисления на социальные нужды | 266,91 |
| 4 Амортизация основных средств и нематериальных активов, используемых в предпринимательской деятельности | 6,75 |
| 5 Прочие затраты | 2431,45 |
| Итого затрат на производство и реализацию продукции | 3534,90 |

**Расчет отпускной цены разработки программного продукта**

Отпускная цена разработки программного продукта без учета НДС Ц, руб., рассчитывается по формуле

Ц = Спол + Пр.

Ц = 3541,50 + 460,40 = 4001,90 руб.

Прибыль рассчитывается по формуле

Пр = ,

где НР – норматив рентабельности , 13%;

Пр = = 460,40 руб.

Цена отпускная с учетом НДС Цотп,руб.,рассчитывается по формуле

Цотп = Ц + НДС,

НДС = ,

где НДС – налог на добавленную стоимость ,руб.;

hндс – ставка налога на добавленную стоимость , %;

НДС =  = 800,38

Цотп = 4001,90+800,38 = 4802,30руб.

Результаты расчетов заносим в таблицу 3

Таблица 3 – Калькуляция отпускной цены разработки программного продукта

|  |  |
| --- | --- |
| Наименование статей калькуляции | Сумма, руб. |
| 1 | 2 |
| 1 Стоимость расходных материалов | 35,95 |
| 2 Стоимость электроэнергии | 15,41 |
| Итого материальные затраты | 51,36 |
| 3 Затраты на оплату труда | 785,02 |
| 4 Отчисления на социальные нужды | 266,91 |
| 5 Амортизация основных средств и нематериальных активов , используемых в предпринимательской деятельности | 6,75 |
| 6 Прочие затраты | 2431,45 |
| Итого полная себестоимость | 3541,50 |
| 7 Прибыль | 460,40 |
| Отпускная цена без учета НДС | 4001,90 |
| 8 Налог на добавленную стоимость | 800,38 |
| Итого отпускная цена с учетом НДС | 4802,30 |

**Расчет технико-экономических показателей**

Эффективность разработки программного продукта подтверждается технико-экономическими показателями:

– трудоемкость разработки программного продукта;

– полная себестоимость;

– прибыль;

– отпускная цена;

– рентабельность продукта;

– материалоемкость;

– дельный вес ТЭР в себестоимости;

–затраты на 1 рубль реализованной продукции.

Рентабельность продукта R %, – показатель оценки эффективного использования текущих затрат на разработку программного продукта и рассчитывается по формуле

R = ×100

R =  ×100=13%

Материалоемкость Ме, руб/руб., – показывает долю материальных затрат в выручке продукции и рассчитывается по формуле

Ме =

Ме =  = 0,10руб./руб.

Удельный вес топливно-энергетических ресурсов в себестоимости УдТЭР %, – показывает долю топливно-энергетических затрат в себестоимости продукции и рассчитывается по формуле

= ×100

=×100 = 0,32%

Затраты на 1 рубль реализованной продукции Зреал,руб/руб., - это один из показателей эффективности производства и определяется по формуле

Зреал =

Зреал = = 0,74руб./руб.

Таблица 4 –Технико-экономические показатели

|  |  |  |
| --- | --- | --- |
| Наименование показателей | Единица измерения | Величина показателя |
| 1 Трудоемкость разработки программного продукта | ч. | 98,82 |
| 2 Полная себестоимость | руб. | 3541,50 |
| 3 Прибыль | руб. | 460,40 |
| 4 Рентабельность программного продукта | % | 13 |
| 5 Отпускная цена изделия с учетом НДС | руб. | 4802,30 |
| 6 Материалоемкость | руб/руб. | 0,10 |
| 7 Удельный вес топливно-энергетических ресурсов в себестоимости | % | 0,32 |
| 8 Затраты на 1 рубль реализованной продукции | руб/руб. | 0,74 |

Отпускная цена программного продукта с учетом НДС составит 4802,30 руб.