**Введение**

Разработка интернет-ресурса для воспитателя ГУО «Детский сад № 102 г. Гродно»

Проект посвящен созданию персонального сайта воспитателя, который будет служить цифровой визитной карточкой и платформой для представления информации о воспитателе, его профессиональной деятельности и образовательной работе. Сайт включает разделы с личной информацией, фотографиями детей, учебно-методическими материалами и документами, доступными для родителей и коллег.

Реализация сайта ориентирована на создание удобного и эстетически привлекательного пользовательского интерфейса, который позволит легко находить нужную информацию. Проект учитывает требования к конфиденциальности данных, особенно в части размещения фотографий детей, и обеспечивает их безопасное хранение. Сайт станет эффективным инструментом для презентации достижений воспитателя и организации взаимодействия с родителями и коллегами.

Состав содержания пояснительной записки

1 анализ задачи

1.1 постановка задачи: определяются ключевые требования к функциональности сайта, включая разделы с личной информацией, фотогалереей, документами и контактами;

1.2 диаграмма вариантов использования: приводятся сценарии взаимодействия пользователей с сайтом: просмотр фотоальбомов, скачивание документов, изучение информации о воспитателе;

1.3 выбор стратегии разработки и модели жизненного цикла\*\*: выбирается модель разработки, учитывающая этапность и итеративный подход;

1.4 инструменты разработки: перечисляются используемые технологии, включая платформу для разработки, язык программирования и системы управления контентом;

1.5 разработка плана работы над проектом: представляется подробный график выполнения задач на всех этапах разработки;

2 проектирование задачи

2.1 разработка структуры сайта, системы меню, навигации: описывается логическая организация сайта, включая основные разделы и элементы навигации;

2.2 разработка UML-диаграмм: создаются диаграммы, отображающие архитектуру системы и потоки данных;

2.3 разработка пользовательского интерфейса: разрабатываются макеты страниц, описываются стилистика и функциональные элементы;

2.4 тест-кейсы: составляются сценарии для проверки корректности работы всех функциональных элементов;

3 реализация

3.1 руководство программиста: приводятся инструкции для установки, настройки и сопровождения сайта, описываются технические решения;

3.2 диаграмма компонентов: демонстрируется взаимодействие между модулями сайта, включая обработку данных и хранение информации;

4 тестирование

Описывается процесс тестирования сайта: проверка работоспособности интерфейса, безопасности данных и отображения страниц на различных устройствах.

5 руководство пользователя

Приводятся инструкции для воспитателя по загрузке фотографий, добавлению документов и обновлению информации.

Заключение

Резюмируются итоги работы, оценивается успешность достижения поставленных целей, а также перспективы развития проекта.

Список использованных источников

Приводится перечень литературы и технической документации, использованной при разработке проекта.

**1 Анализ задачи**

**1.1 Постановка задачи**

**1.1.1 Организационно-экономическая сущность задачи**

- наименование задачи: Разработка персонального сайта воспитателя ГУО «Детский сад № 102 г. Гродно»;

- цель разработки: Создание веб-ресурса, который служит платформой для предоставления информации о воспитателе, публикации фотографий детей, учебных материалов и документов, а также для взаимодействия с родителями;

- назначение: Сайт предназначен для воспитателей, родителей воспитанников и сотрудников образовательного учреждения, а также для потенциальных заинтересованных лиц;

- периодичность использования: Персональный сайт используется регулярно, как для оперативного обмена информацией, так и для постоянного хранения и обновления образовательных материалов;

- источники и способы получения данных: Информация вводится вручную воспитателем (например, фотографии, документы) или предоставляется администрацией детского сада;

- информационная связь с другими задачами: Сайт интегрируется с локальными информационными системами детского сада для обеспечения согласованности данных. Также возможна связь с электронными почтовыми сервисами для уведомлений.

- обзор существующих аналогичных ПП: В сети Интернет можно найти множество сайтов подобного формата, предназначенных для презентации деятельности воспитателей. На таких ресурсах обычно размещается информация о профессиональной деятельности воспитателя, его достижениях и квалификации. Одним из ключевых элементов подобных сайтов являются фотогалереи, где представлены моменты из жизни детей: занятия, праздники, прогулки и другие мероприятия;

**1.1.2 Функциональные требования**

Персональный сайт должен выполнять следующие функции:

- размещение личной информации о воспитателе (биография, образование, достижения);

- публикация фотографий воспитанников и событий в виде галереи с возможностью ограниченного доступа;

- хранение и предоставление методических материалов и образовательных документов;

**1.1.3 Описание исходной (входной) информации**

- личная информация воспитателя (ФИО, образование, контактные данные);

- фотографии воспитанников и мероприятий;

- учебно-методические материалы и документы (в формате PDF, Word);

**1.1.4 Описание результатной (выходной) информации**

**1.1.4.1 Перечень результатной информации:**

- веб-страницы с размещенной информацией;

- галереи фотографий;

- документы для скачивания;

**1.1.4.2 Формы представления:**

- отображение информации в браузере;

- скачивание файлов;

**1.1.4.3 Периодичность и сроки представления:**

- информация обновляется по мере необходимости;

**1.1.4.4 Пользователи результатной информации:**

- Родители воспитанников, коллеги воспитателя.

**1.1.5 описание используемой условно-постоянной информации**

**1.1.5.1 Перечень условно-постоянной информации:**

- контактные данные детского сада;

- доступные учебно-методические материалы;

**1.1.5.2 Форма представления:**

- справочники, списки;

**1.1.6 Нефункциональные (эксплуатационные) требования**

- требования к применению: Сайт должен быть удобен в использовании, адаптирован под мобильные устройства, сопровождаться пользовательской инструкцией;

- требования к производительности: Время загрузки страниц не превышает 3 секунд;

- требования к реализации: Поддержка всех современных браузеров;

- требования к надежности: Должна быть реализована регулярная архивация данных; система должна устойчиво работать при сбоях;

- требования к интерфейсу: Интуитивный интерфейс с понятной навигацией для всех категорий пользователей;

**1.2 Диаграмма вариантов использования**

Посетитель сайта может просматривать информацию о воспитателе, открывать методические материалы для ознакомления и просматривать фотографии в фотогалерее.

Воспитатель имеет возможность добавлять новые фотографии, загружать методические материалы и изменять информацию о себе для поддержания актуальности данных. Разработанная диаграмма вариантов использования представлена в приложении А.

**1.3 Выбор стратегии разработки и модели жизненного цикл**

Для разработки интернет ресурса для воспитателя ГУО «Детский сад № 102 г. Гродно» следует выбрать стратегию разработки и модель жизненного цикла. Осуществляем выбор посредством составления таблиц:

Таблица 1 **–** Выбор модели жизненного цикла на основе характеристик требований

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории требований | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |

Продолжение таблицы 1

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1. | Являются ли требования к проекту легко определимыми и реализуемыми? | Да | Да | Да | Нет | Нет | Нет |
| 2. | Могут ли требования быть сформулированы в начале ЖЦ? | Да | Да | Да | Да | Нет | Нет |
| 3. | Часто ли будут изменяться требования на протяжении ЖЦ? | Нет | Нет | Нет | Нет | Да | Да |
| 4. | Нужно ли демонстрировать требования с целью их определения? | Нет | Нет | Да | Нет | Да | Да |
| 5. | Требуется ли проверка концепции программного средства или системы? | Нет | Нет | Да | Нет | Да | Да |
| 6. | Будут ли требования изменяться или уточняться с ростом сложности системы (программного средства) в ЖЦ? | Нет | Нет | Нет | Да | Да | Да |
| 7. | Нужно ли реализовать основные требования на ранних этапах разработки? | Нет | Нет | Да | Да | Да | Да |

Вычисления: 5 за каскадную, 5 за V- образную, 5 за RAD, 4 за инкрементную 2 за быстрого прототипирования и 2 за эволюционную.

Итог: На основе результатов заполнения табл. 1 подходящей является каскадная модель, V- образная и RAD модель.

Таблица 2 **–** Выбор модели жизненного цикла на основе характеристик команды разработчиков

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории команды разработчиков  проекта | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1. | Являются ли проблемы предметной области проекта новыми для большинства разработчиков? | Нет | Нет | Нет | Нет | Да | Да |
| 2. | Являются ли инструментальные средства, используемые в проекте, новыми для большинства разработчиков? | Да | Да | Нет | Нет | Нет | Да |

Продолжение таблицы 2

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 3. | Изменяются ли роли участников проекта на протяжении ЖЦ? | Нет | Нет | Нет | Да | Да | Да |
| 4. | Является ли структура процесса разработки более значимой для разработчиков, чем гибкость? | Да | Да | Нет | Да | Нет | Нет |
| 5. | Важна ли легкость распределения человеческих ресурсов проекта? | Да | Да | Да | Да | Нет | Нет |
| 6. | Приемлет ли команда разработчиков оценки, проверки, стадии разработки? | Да | Да | Нет | Да | Да | Да |

Вычисления: 5 за каскадную, 5 за V-образную, 4 за RAD, 5 за инкрементную, 2 за быстрого прототипирования и 3 за эволюционную.

Итог: На основе результатов заполнения табл. 2 подходящими являются каскадная, V-образная и инкрементная модели.

Таблица 3 **–** Выбор модели жизненного цикла на основе характеристик коллектива пользователей

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | | Критерии категории коллектива пользователей | | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1. | Будет ли присутствие пользователей ограничено в ЖЦ разработки? | | Да | | Да | Нет | Да | Нет | Да |
| 2. | Будут ли пользователи оценивать текущее состояние программного продукта (системы) в процессе разработки? | | Нет | | Нет | Нет | Да | Да | Да |
| 3. | Будут ли пользователи вовлечены во все фазы ЖЦ разработки? | | Нет | | Нет | Да | Нет | Да | Нет |
| 4. | Будет ли заказчик отслеживать ход выполнения проекта? | | Нет | | Нет | Нет | Нет | Да | Да |

Вычисления: 0 за каскадную, 0 за V-образную, 2 за RAD, 1 за инкрементную, 4 за быстрого прототипирования и 2 за эволюционную.

Итог: На основе результатов заполнения табл. 3 подходящей является модель быстрого проектирования.

Таблица 4 **–** Выбор модели жизненного цикла на основе характеристик типа проектов и рисков

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | | Критерии категории типов проекта и рисков | | Каскадная | | V-образная | | RAD | | Инкрементная | | Быстрого прототипирования | | Эволюционная | |
| 1. | | Разрабатывается ли в проекте продукт нового для организации направления? | | Нет | | Нет | | Нет | | Да | | Да | | Да | |
| 2. | | Будет ли проект являться расширением существующей системы? | | Да | | Да | | Да | | Да | | Нет | | Нет | |
| 3. | | Будет ли проект крупно- или среднемасштабным? | | Нет | | Нет | | Нет | | Да | | Да | | Да | |
| 4. | | Ожидается ли длительная эксплуатация продукта? | | Да | | Да | | Нет | | Да | | Нет | | Да | |
| 5. | | Необходим ли высокий уровень надежности продукта проекта? | | Нет | | Да | | Нет | | Да | | Нет | | Да | |
| 6. | | Предполагается ли эволюция продукта проекта в течение ЖЦ? | | Нет | | Нет | | Нет | | Да | | Да | | Да | |
| 7. | | Велика ли вероятность изменения системы (продукта) на этапе сопровождения? | | Нет | | Нет | | Нет | | Да | | Да | | Да | |
| 8. | | Является ли график сжатым? | | Нет | | Нет | | Да | | Да | | Да | | Да | |
| 9. | | Предполагается ли повторное использование компонентов? | | Нет | | Нет | | Да | | Да | | Да | | Да | |
| 10. | | Являются ли достаточными ресурсы (время, деньги, инструменты, персонал)? | | Нет | | Нет | | Нет | | Нет | | Да | | Да | |

Вычисления: 3 за каскадную, 4 за V-образную, 4 за RAD, 8 за инкрементную, 6 за быстрого прототипирования и 8 за эволюционную.

Итог: На основе результатов заполнения табл. 4 подходящей является инкрементная и эволюционная модели.

Общий итог: в итоге заполнения табл. 1 – 4 наиболее подходящей является инкрементная модель.

**1.4 Инструменты разработки**

Для разработки данного проекта будет выбрана среда разработки Visual Studio, которая является наиболее актуальной средой для создания приложений данного типа.

Разработка будет производится на таких языках программирования, как:

- HTML (HyperText Markup Language) – для создания разметки сайта;

- CSS (Cascading Style Sheets) – формальный язык описания внешнего вида документа, написанного с использованием языка разметки;

- JavaSсript – «оживление сайта», задание реакции элементов на действия пользователя; прототипно-ориентированный сценарный язык программирования. JavaScript обычно используется как встраиваемый язык для программного доступа к объектам приложений. Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности веб-страницам.

Иные инструменты, используемые при разработке и написании сопутствующей документации:

- DRAW.IO – будет использоваться для создания графической части и разработки UML-диаграмм;

- Microsoft Office Word – для написания документации к программному продукту;

- PhpMyAdmin 5.2.1 – будет использоваться в качестве инструмента для администрирования СУБД MySQL;

- OpenServer 6.0.0 – средство для разработки интернет-приложений. OpenServer содержит набор дистрибутивов (Apache+SSL, PHP5 в виде модуля, MySQL, phpMyAdmin и т.д.) и является программной оболочкой, используемой Web-разработчиками (программистами и дизайнерами) для отладки сайтов на локальной Windows-машине без необходимости выхода в Интернет;

Разработка проекта будет происходить на компьютере со следующими параметрами:

- процессор Intel® Core i3-8100 CPU @ 3.60GHz;

- объем оперативной памяти 16 GB;

- объем места на жестком диске 1 TB;

- видеокарта NVIDIA ® GTX 1080 with 8 GB VRAM;

- ОС Windows 10 Pro

**1.5 Разработка плана работы над проектом**

Назначение диаграммы Ганта

Диаграмма Ганта предназначена для планирования, управления и контроля выполнения задач в рамках проектов. Она представляет собой графическое отображение плана работы, где ось X показывает временные рамки, а ось Y — список задач или этапов. Каждый элемент диаграммы отображается в виде горизонтальной полосы, длина которой соответствует продолжительности выполнения задачи.

Диаграмма Ганта будет представлена в приложении Б.

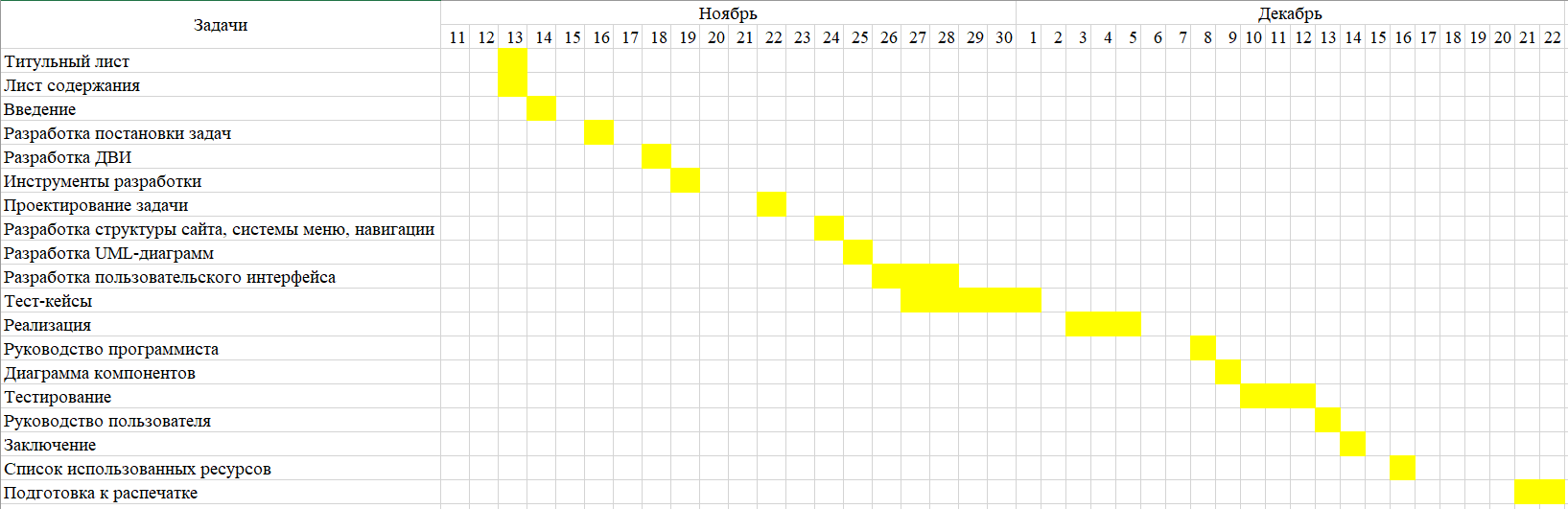


Рисунок 1 — Диаграмма Ганта

**2 Проектирование задачи**

**2.1 Разработка структуры сайта, системы меню, навигации**

Диаграмма, отражающая структуру сайта, систему меню и навигацию, предназначена для визуализации логической организации веб-ресурса и связей между его элементами. Она помогает проектировщику и разработчику четко представить, как будет организована информация на сайте и как пользователи смогут перемещаться между его разделами. Разработанная диаграмма структуры сайта, системы меню, навигации представлена в приложении В.

**2.2 Разработка UML-диаграмм**

**2.2.1 Диаграмма деятельности**

Диаграмма деятельности – это UML - диаграмма, на которой показаны действия, состояния которых описано на диаграмме состояний. Диаграммы деятельности предоставляют графическое представление процессов, что позволяет команде разработки и заинтересованным сторонам легко понять, как осуществляется работа. Это особенно полезно для сложных процессов, где трудно отслеживать последовательность действий.

В данной диаграмме представлены функции, которые может выполнять пользователь и администратор. Диаграмма деятельности представлена в приложении Г.

**2.2.2 Диаграмма последовательности**

Диаграмма последовательности — UML-диаграмма, на которой для некоторого набора объектов на единой временной оси показан жизненный цикл объекта и взаимодействие актеров информационной системы в рамках прецедента.

Диаграммы последовательностей используются для уточнения диаграмм прецедентов, более детального описания логики сценариев использования. Это отличное средство документирования проекта с точки зрения сценариев использования.

Диаграммы последовательностей обычно содержат объекты, которые взаимодействуют в рамках сценария, сообщения, которыми они обмениваются, и возвращаемые результаты, связанные с сообщениями. Впрочем, часто возвращаемые результаты обозначают лишь в том случае, если это не очевидно из контекста.

Объекты обозначаются прямоугольниками с подчеркнутыми именами (чтобы отличить их от классов).

Сообщения (вызовы методов) - линиями со стрелками.

Возвращаемые результаты - пунктирными линиями со стрелками.

Прямоугольники на вертикальных линиях под каждым из объектов показывают «время жизни» (фокус) объектов. Впрочем, довольно часто их не изображают на диаграмме, все это зависит от индивидуального стиля проектирования.

Диаграмма последовательности будет представлена в приложении Д. На данной диаграмме показан процесс скачивания файлов.

**2.2.3 Диаграмма компонентов**

Диаграмма компонентов (Component Diagram) — это один из типов UML-диаграмм, который отображает структуру системы на уровне физических и логических компонентов. Компонентами могут быть модули, классы, библиотеки, файлы, пакеты, базы данных или веб-сервисы. Диаграмма показывает, как эти компоненты связаны между собой и какие интерфейсы используются для взаимодействия.

Данная диаграмма представлена в приложении Е.

**2.3 Разработка пользовательского интерфейса**

**2.3.1 UX – прототипы**

UX-прототипы — это визуальные макеты или интерактивные модели пользовательского интерфейса, предназначенные для проектирования и тестирования пользовательского опыта (UX, User Experience) до начала разработки программного продукта. Они отображают структуру и функциональность будущего интерфейса, помогая понять, как пользователи будут взаимодействовать с продуктом. Прототипы могут быть как статическими (простыми схемами), так и интерактивными (с возможностью кликов, навигации и выполнения определённых действий).

Для чего нужны UX-прототипы

**-** проектирование интерфейса:

UX-прототипы помогают визуализировать и продумать структуру будущего продукта, включая расположение элементов интерфейса, логику навигации и взаимодействия.

- тестирование пользовательского опыта:

Прототипы позволяют проводить тестирование с реальными пользователями, чтобы выявить неудобные элементы интерфейса и доработать их ещё до начала программирования.

- согласование с заказчиком:

UX-прототипы являются отличным инструментом для обсуждения и согласования структуры и функционала продукта с заказчиком, упрощая процесс внесения изменений.

- экономия времени и ресурсов:

Использование прототипов позволяет сократить затраты на исправление ошибок, которые могли бы возникнуть на этапе разработки из-за плохо продуманного интерфейса.

- улучшение коммуникации в команде:

Прототипы служат общей основой для обсуждения между дизайнерами, разработчиками, менеджерами и другими членами команды.

- визуализация идеи:

Прототипы делают идею более наглядной, помогая быстро показать, как будет выглядеть и работать продукт.

Для данного веб-сайта показаны примеры главной страницы, обо мне, методической копилки, фотогалерея, мнение о сайте

Разработанные UX-прототипы для данного веб-сайта можно просмотреть по ссылке: https://www.figma.com/design/VIUb4Ik7NrOphjtmQhmDv6/UX?node-id=0-1&t=FJ69hxksiVkL0O6Z-1

С прототипами UX вы можете ознакомиться в приложении Ж.

**2.3.2 UI – прототипы**

UI-прототипы — это визуальные макеты пользовательского интерфейса (UI, User Interface), которые демонстрируют внешний вид и дизайн будущего программного продукта. Они фокусируются на графической составляющей: расположении элементов, цветовой палитре, шрифтах, иконках, анимациях и других визуальных деталях. UI-прототипы служат для проработки эстетической части интерфейса и проверки, насколько он соответствует задачам и ожиданиям пользователей.

Для чего нужны UI-прототипы

- визуализация дизайна:

UI-прототипы позволяют увидеть, как будет выглядеть продукт, включая его стилистику, графику и оформление.

- согласование дизайна с заказчиком:

Прототипы помогают согласовать с заказчиком внешний вид интерфейса на этапе разработки, что снижает вероятность переделок в будущем.

- упрощение работы разработчиков:

UI-прототипы дают разработчикам точное представление о том, как должен выглядеть конечный продукт, включая размеры, шрифты, цвета и расположение элементов.

- проверка визуальной эстетики:

Прототипы позволяют протестировать, насколько визуальная часть интерфейса соответствует бренду, целевой аудитории и задачам продукта.

- улучшение взаимодействия с пользователем:

Проработка UI-прототипа позволяет создать понятный и интуитивный интерфейс, который делает использование продукта более комфортным для пользователя.

- оптимизация дизайна:

UI-прототипы дают возможность быстро оценить различные варианты дизайна, выбрать лучший и внести изменения до начала этапа программирования.

Разработанные UI-прототипы для данного веб-сайта можно просмотреть по ссылке: https://www.figma.com/design/uJ1YyQf1rcbJi7ehJnYzdH/UI?node-id=0-1&t=qtqSYb2bbZDRmnuo-1

С прототипом UI вы можете ознакомиться в приложении З.

**3 Реализация**

**3.1 Руководство программиста**

Программный продукт разработан с использованием конструктора WordPress 6.0.0. WordPress предлагает интуитивно понятный интерфейс, что делает его доступным даже для пользователей без технических навыков. Установка и управление контентом не требуют глубоких знаний программирования. Существуют тысячи бесплатных и платных тем и плагинов, которые позволяют настраивать внешний вид сайта и добавлять функциональность. Темы определяют дизайн, а плагины расширяют возможности

**3.1.1 Создание сайта**

Для того чтобы создать свой веб-ресурс в WordPress необходимо сначала зарегистрироваться, создать свою учетную запись и подключить базу данных. Для создания своей учетной записи вам необходимо зайти на сайт phpMyAdmin. Чтобы подключить базу данных достаточно придумать ее название и пароль. При регистрации в самом WordPress также надо ввести имя пользователя и придумать ему пароль.

После ввода этих данных вы перейдете в административную панель WordPress, где сможете настраивать свой сайт, устанавливать темы и плагины, а также добавлять контент.

При создании сайта использовал плагины такие как:

Elementor (это популярный конструктор страниц, который позволяет пользователям создавать и редактировать страницы WordPress с помощью перетаскивания элементов);

Envato Elements (это набор дополнений для Elementor, который расширяет его функциональность);

Image Optimizer - Compress, Resize and Optimize Images (это простой в использовании плагин для WordPress, который уменьшает размер изображения с помощью различных методов сжатия и изменения размера, сохраняя при этом максимально возможное качество изображения. Image Optimizer поддерживает множество популярных форматов, включая JPEG, PNG, WebP, AVIF и GIF);

Ocean Extra (добавляет дополнительные функции и гибкость в тему OceanWP);

WPForms Lite (это простой и функциональный конструктор форм WordPress с возможностью перетаскивания. Создавайте контактные формы, а также формы обратной связи, подписки, платежей (в том числе через Stripe, Square и PayPal) и другие типы форм для вашего сайта за считанные минуты, а не часы!).

**3.1.2 Настройка функциональности и дизайна**

Все элементы, видимые на экране, можно изменить, переместить или удалить. При нажатии на кнопку «Редактировать с помощью Elementor» слева экрана появляется панель инструментов редактора, которая состоит из 4 разделов, которые доступны нам в бесплатной версии данного редактора:

1 Макет — этот раздел позволяет изменять структуру раздела сайта;

2 Базовый — этот раздел позволяет добавлять текст, фото, видео, кнопку и так далее;

3 Общие — этот раздел отвечает за добавления блока с фотографиями, слайд бара, галерею, счётчик, прогресс, отзывы предупреждение о чём-либо, музыкальной составляющей и текста по контуру чего-либо;

4 WordPress — этот раздел отвечает за основные разделы самого WordPress;

**3.1.3 Редактирование страниц**

Первым делом, после открытия шаблона(темы) для будущего веб-ресурса, нужно изменить его название. Для этого необходимо войти в админ панель сайта, на панели инструментов слева найдите раздел «Настройки» и выберите «Общие». На странице "Общие настройки" вы увидите два поля, где нужно изменить название сайта.

Теперь следует разобраться с настройкой меню сайта. Тема «Сайт воспитателя» имеет ненужные страницы, поэтому оставляем только главную, обо мне, методическая копилка, фотогалерея, мнение о сайте. Далее чтобы создать остальные разделы перейдите в раздел "Внешний вид": В левой боковой панели выберите "Внешний вид", а затем нажмите на "Меню". Создайте новое меню (введите его название, нажмите «Создать меню»). Добавьте элементы меню: выберите страницы, записи, категории или пользовательские ссылки, которые вы хотите добавить в меню, из ресурсов, представленных слева, установите галочки рядом с элементами, которые хотите добавить, и нажмите "Добавить в меню".

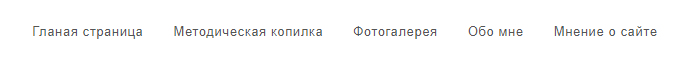


Рисунок 1 — Меню сайта

Хедер данного сайта будет содержать надпись «Сайт воспитателя», горизонтальное меню сайта. Хедер редактировался с помощью встроенных функций WordPress

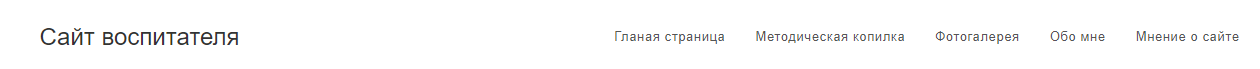


Рисунок 2 — Хедер сайта

Для того чтобы настроить хедер сайта, необходимо зайти во вкладку "Внешний вид", затем выбрать "Меню", добавить страницы и расположить их в таком порядке, чтобы это было удобно для вас.

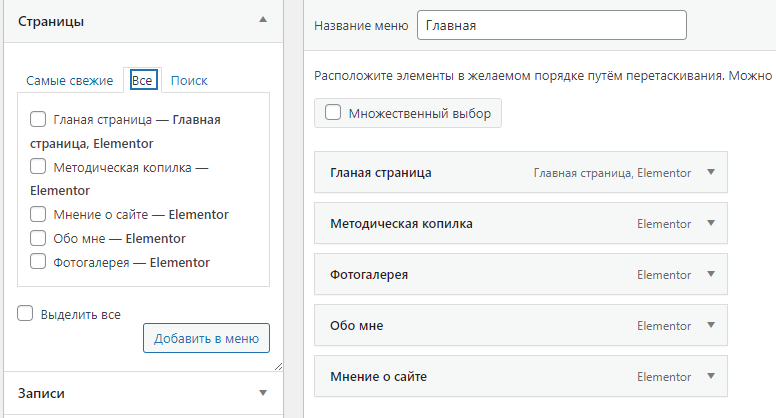


Рисунок 3 — Настройка хеддер

Далее переходим к настройке главной страницы. Основная настройка главной страницы производиться с помощью Elementor.

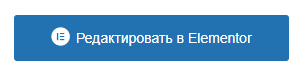


Рисунок 4 — Elementor

После этого в левой части экрана появится набор элементов, которые можно добавить и после их редактировать как нужно.

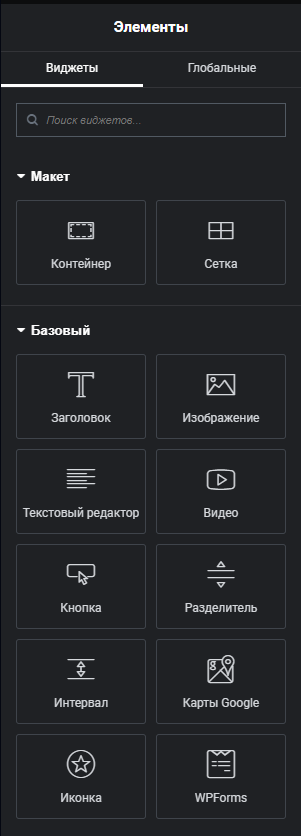


Рисунок 5 — Набор элементов

Мнение о сайте будет приниматься в виде теста Google Forms. Чтобы вставить Google Form на страницу сайта, необходимо после её создания скопировать HTML-ссылку и вставить её в заранее добавленный элемент WordPress HTML-код.

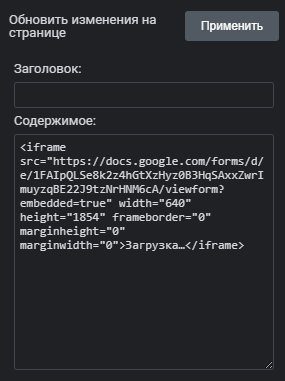


Рисунок 6 — Вставка кода на сайте

**4 Тестирование**

Тестирование является одним из самых важнейших этапов при создании программного продукта. Исходя из внедрения программы, при определённым наборе текстов нельзя сделать однозначных выводов о том, что программа будет исправно работать в любой момент использования. Опираясь на данный набор тестов, можно говорить о некоторой степени уверенности в правильности работы программного продукта.

Таким образом, основная часть ошибок и недоработок была выявлена на этапе реализации проекта. После завершения этапа реализации, было проведено тщательное тестирование.

Программный продукт был протестирован разработчиком, а также заказчиком.

Таблица 5 — Отчёт результатах тестирования

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Тест | Ожидаемый результат | Физический результат | Результат тестирования |
| 1 | Проверка перехода через меню на другие страницы сайта | Открытие других страниц сайта | Открытие других страниц сайта | Выполнено |
| 2 | Проверка читаемости текста | Текст читаемый | Текст читаемый | Выполнено |
| 3 | Проверка просмотра фото | Фото отображается | Фото отображается | Выполнено |
| 4 | Проверка просмотра документов | Документы просматриваются успешно | Документы просматриваются успешно | Выполнено |
| 5 | Проверка работоспособности кнопки скачивания документа | При нажатии на кнопку документ начинает загружаться | При нажатии на кнопку документ начинает загружаться | Выполнено |
| 6 | Проверка работоспособности изменения администратором | Администратор может добавлять или изменять информацию | Администратор может добавлять или изменять информацию | Выполнено |
| 7 | Проверка проходимости теста на странице мнение о сайте | Тест google forms успешно проходится | Тест google forms успешно проходится | Выполнено |

В ходе тестирования были выявлены ошибки разной степени важности. Все обнаруженные дефекты были устранены, после чего программный продукт прошёл проверку на устойчивость к нестандартным сценариям использования, а также на качество реализации логики и её применения.

В результате можно заключить, что программный продукт готов к использованию без значительных проблем или неудобств для пользователей.

Тест-кейсы представлены в приложении И.

**5 Применение**

**5.1 Общие сведения о программном продукте**

Цель данного проекта – создать удобный интернет-ресурс для воспитателя, который будет служить платформой для предоставления информации, обмена опытом и взаимодействия с родителями и коллегами. В процессе разработки решаются различные задачи, способствующие не только реализации проекта, но и углублению знаний в области веб-разработки и дизайна.

Разрабатываемый сайт предназначен для воспитателей, родителей, коллег и всех, кто интересуется образовательной и воспитательной деятельностью. Он будет полезен как профессионалам, так и тем, кто стремится получить больше информации об этой сфере.

**5.2 Руководство пользователю**

Для того, чтобы открыть сайт необходимо перейти по ссылке. Затем достаточно следовать приведенной инструкции установки приложения. После открытия сайта нас встречает главная страница, где в меню необходимо выбрать интересующую категорию.

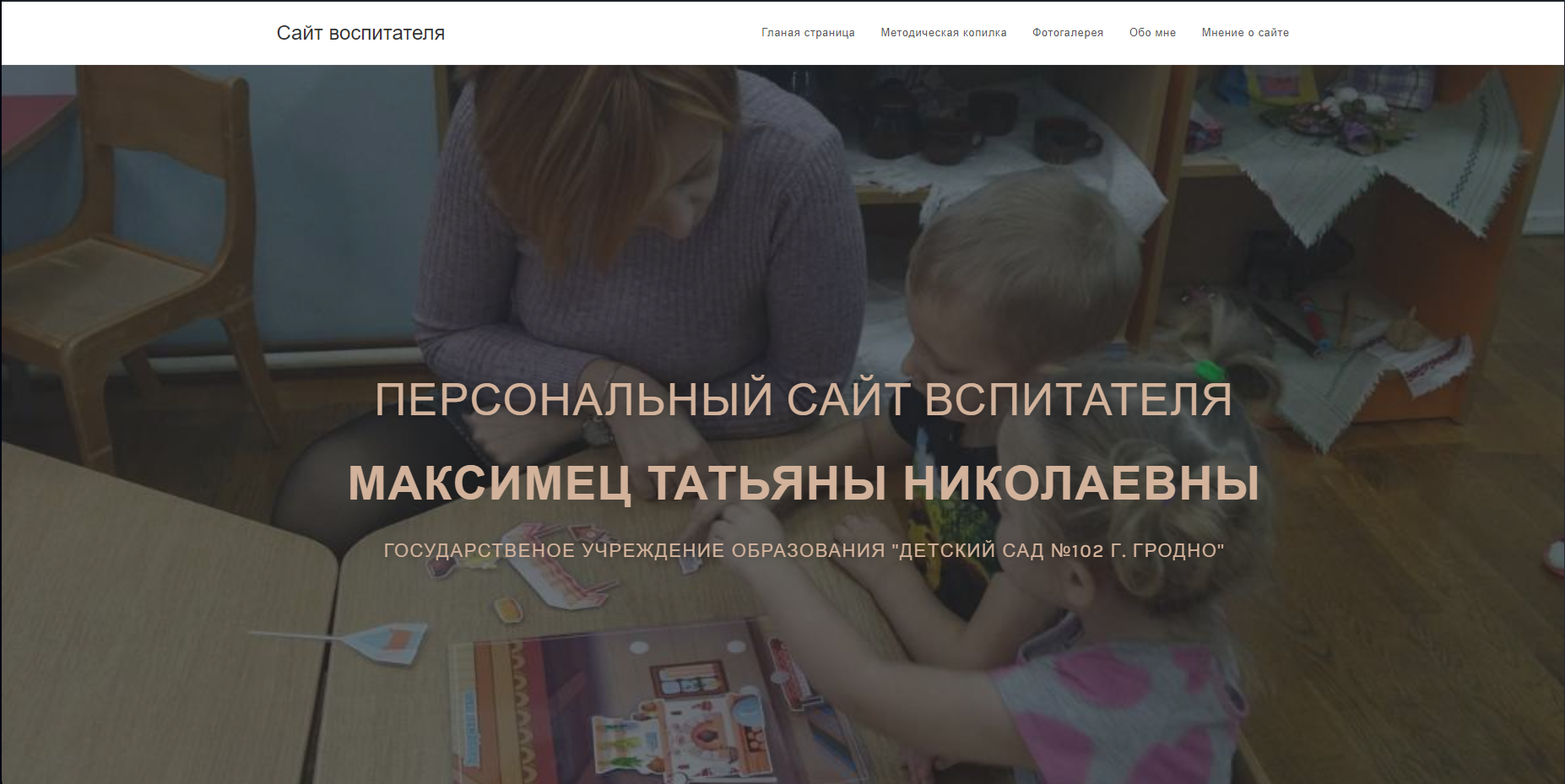


Рисунок 7 — Главная страница сайта

Промотав страницу немного вниз, можно найти надпись и кнопку, с помощью которой можно перейти на другую страницу.

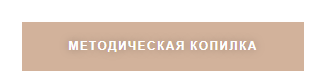


Рисунок 8 — Кнопка с переходом на другую страницу

Допустим, после перехода на страницу "Обо мне" здесь представлены: фото воспитателя, должность, педагогический стаж, место учёбы и небольшое эссе на тему «Почему я решила стать воспитателем».

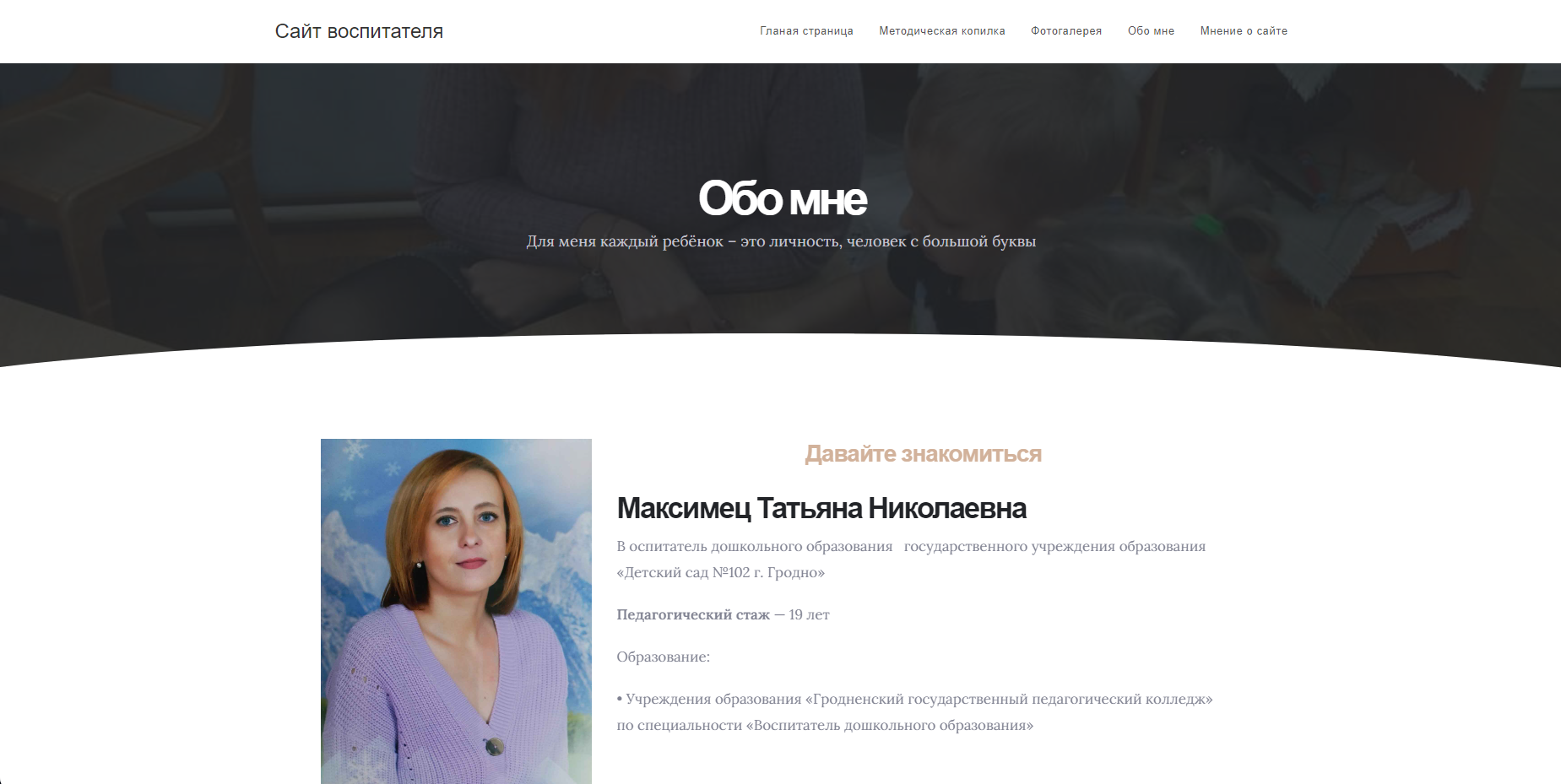


Рисунок 9 — Страница обо мне

В самом низу каждой страницы расположена кнопка для перехода на главную страницу.



Рисунок 10 — Кнопка перехода на главную

На странице «Методическая копилка» представлены 5 советов для родителей, а также 5 документов для педагогов.

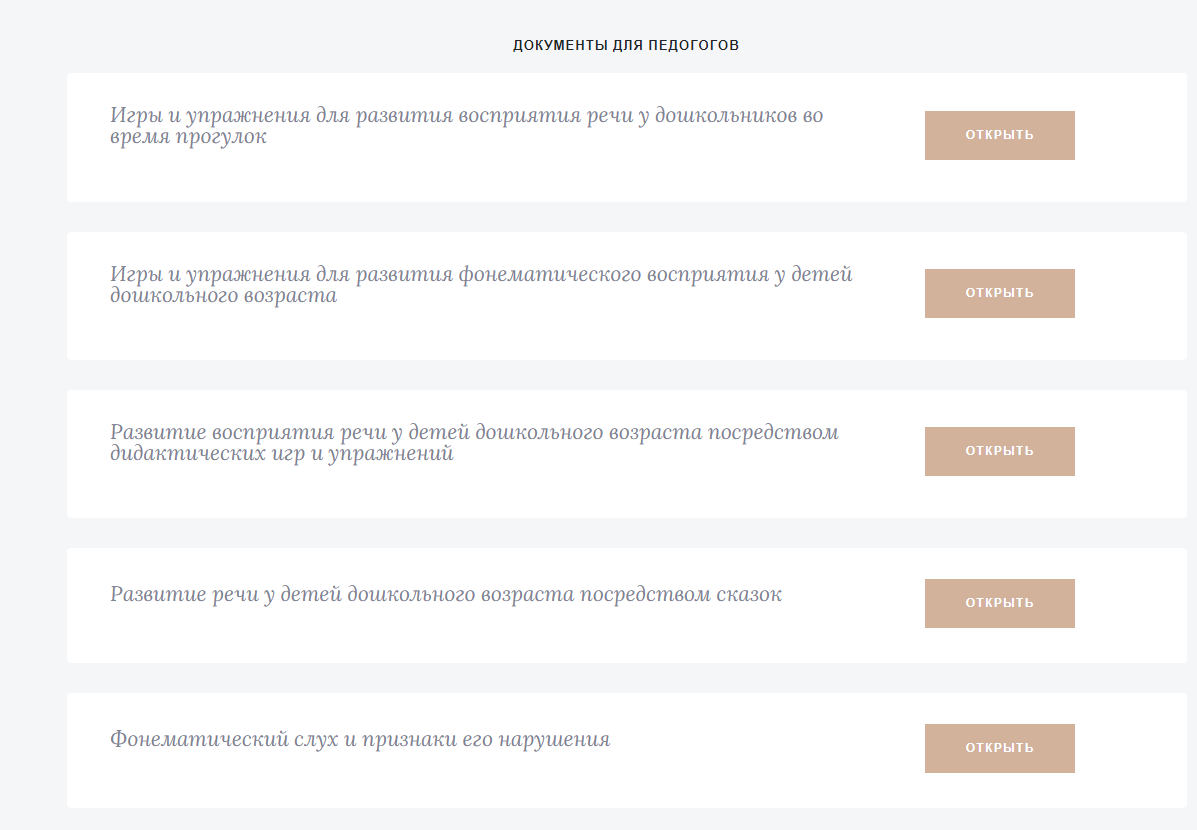


Рисунок 11 — Документы для педагогов

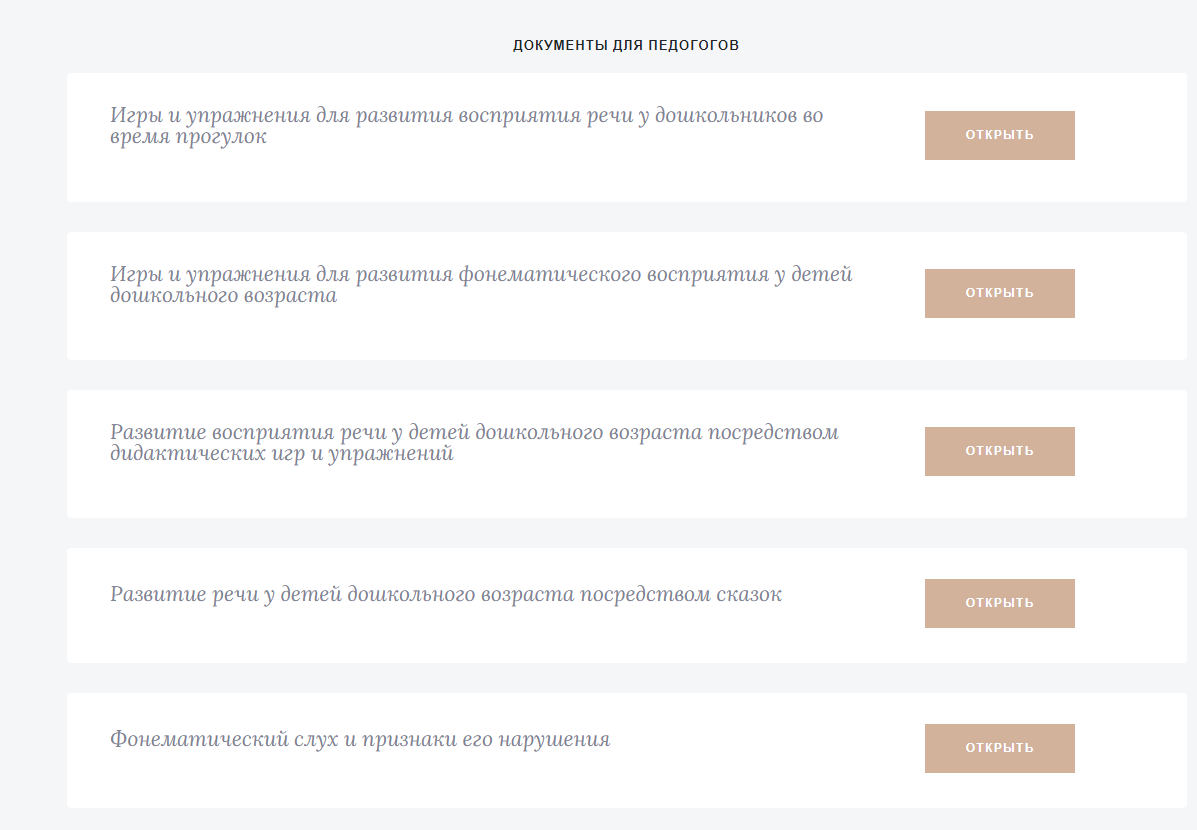


Рисунок 12 — Советы для родителей

Для того чтобы открыть документы, необходимо нажать на кнопку «Открыть».



Рисунок 12 — Кнопка «открыть»

После этого откроется документ, сохранённый на Google Диске.

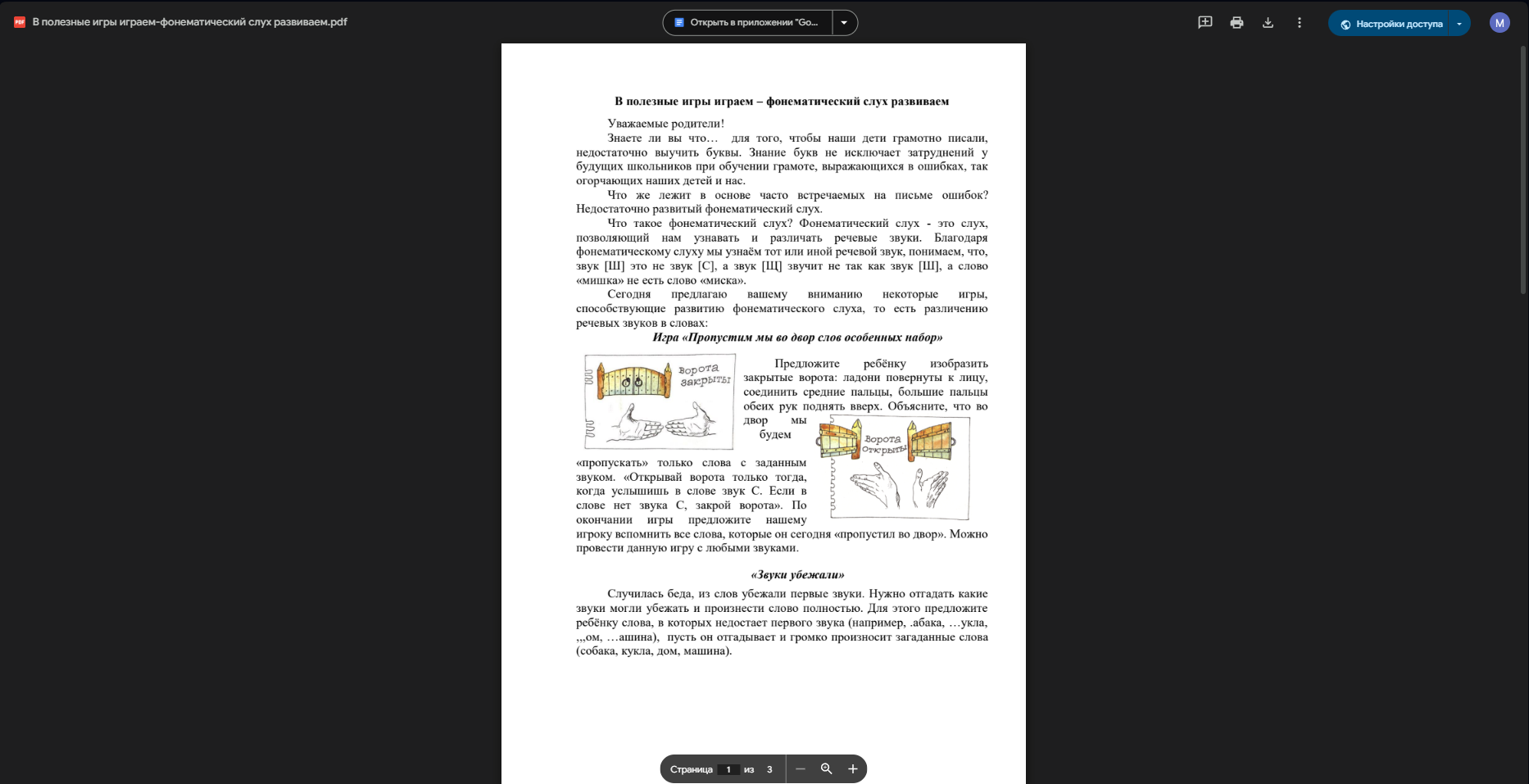


Рисунок 13 — Документ

Для того чтобы скачать данный документ, необходимо нажать на кнопку, представленную ниже.



Рисунок 14 — Кнопка для скачивания файла

На странице под названием «Фотогалерея» вы сможете увидеть фото детей, пролистав немного ниже.



Рисунок 15 — Фото данной страницы

Осталась последняя страница под названием «Мнение о сайте».

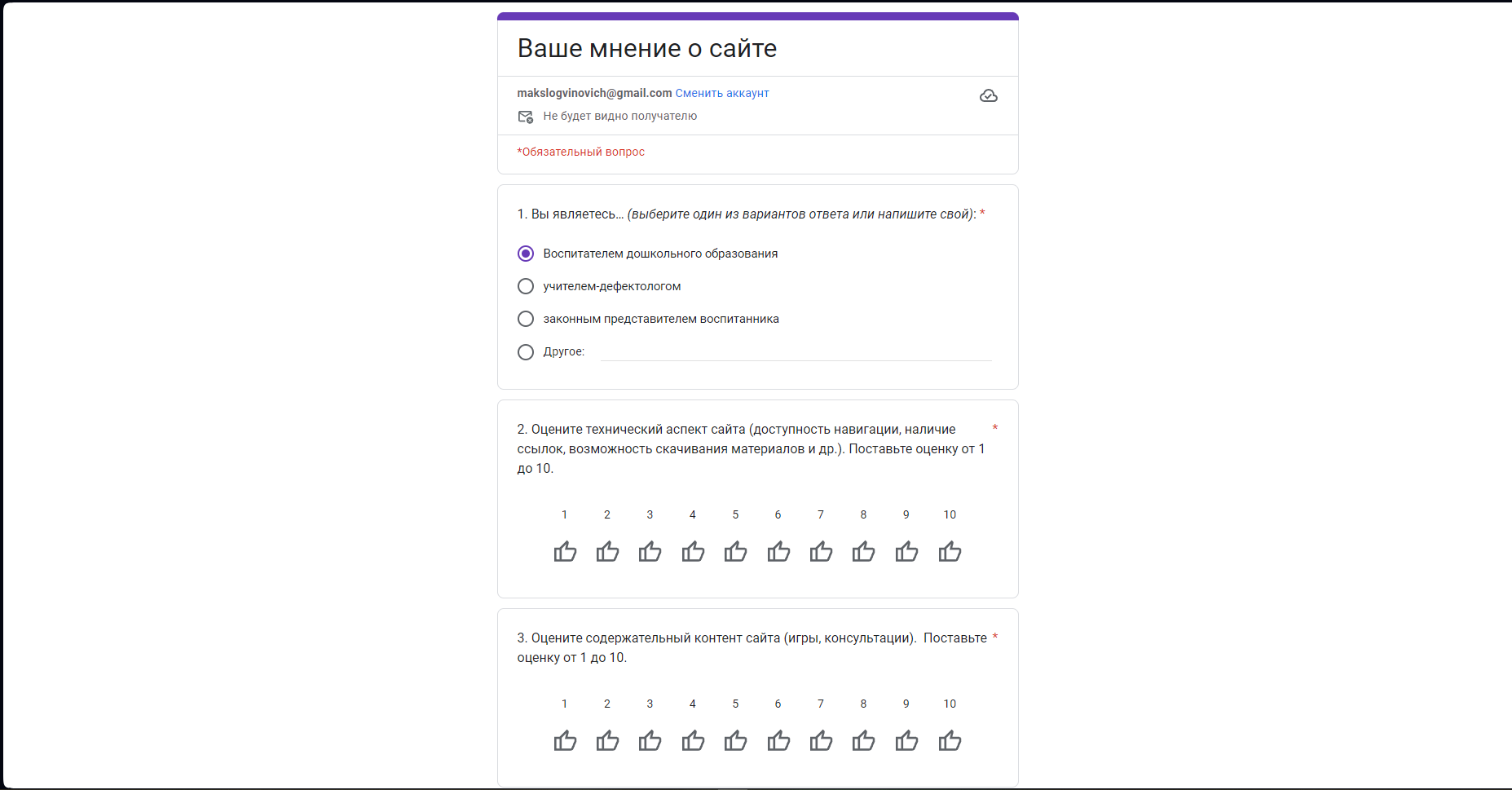


Рисунок 16 — Фото этой страницы

**Заключение**

В ходе выполнения курсового проекта была разработана интернет-платформа для воспитателя ГУО «Детский сад № 102 г. Гродно», которая предоставляет удобные инструменты для взаимодействия с родителями и коллегами, а также для демонстрации профессиональной деятельности воспитателя.

В рамках работы были проанализированы требования к системе, спроектирована структура сайта, разработан пользовательский интерфейс, а также реализованы основные функциональные модули. Среди ключевых страниц сайта можно выделить:

- «Обо мне», содержащую информацию о воспитателе и его профессиональном пути;

- «Методическую копилку», где представлены полезные советы для родителей и документы для педагогов;

- «Фотогалерею» с фотографиями детей;

- страницу «Мнение о сайте», которая позволяет собирать отзывы пользователей с помощью интеграции Google Forms;

Все этапы разработки сопровождались тестированием, в ходе которого были выявлены и устранены ошибки, что позволило обеспечить стабильную и корректную работу сайта. Реализованный продукт соответствует целям проекта: он удобен в использовании, функционален и может быть успешно применён в профессиональной деятельности воспитателя.

Данный проект позволил углубить знания в области веб-разработки, проектирования пользовательского интерфейса и работы с современными инструментами создания сайтов. Полученные результаты подтверждают, что разработанный ресурс способен эффективно решать поставленные задачи и приносить практическую пользу пользователям.

**Список использованных источников**

1 Как создать сайт на WordPress [Электронный ресурс]. URL: https://info/blog/site-on-wordpress/ – Дата доступа: 30.11.2024.

2 Официальная документация Google Forms [Электронный ресурс]. URL: https://www.google.com/forms/about/ Дата доступа: 15.12.2024.