**Введение**

Первый раздел носит название «Анализ задачи». В нем вы сможете ознакомиться с постановкой задачи, включающей исследование предметной области. Также в разделе будут описаны входные данные (например, логин пользователя, пароль пользователя) и выходные данные (учебный материал, данные об успеваемости). В подразделе «Инструменты разработки» будет рассмотрена выбранная среда программирования.

В разделе «Проектирование задачи» будут рассмотрены основные аспекты разработки программы. Описывается структура хранения данных (например, база данных сайта) и способы их обработки. Здесь можно узнать об организации данных в контексте среды разработки. В данном разделе будет описан пользовательский интерфейс, составлены алгоритмы процесса обработки информации.

«Реализация» – это третий раздел отчета, в котором описываются все элементы и объекты, которые будут использованы при реализации данного приложения.

Четвертый раздел – «Тестирование». В этом разделе будет описано тестирование функционала программы. Проверка охватит ввод, обработку и вывод данных, а также корректность выполнения пользовательских операций. Будут смоделированы все возможные действия пользователя при работе с web-ресурсом, начиная от входа на сайт заканчивая закрытием вкладки.

В разделе «Руководство пользователя» будет описано назначение, область применения, среда функционирования данного программного продукта.

«Заключение» будет содержать краткую формулировку задачи, результаты проделанной работы, описание использованных методов и средств.

В разделе «Список использованных источников» будет приведен список источников, использованных при разработке программы.

В приложении А будет представлена диаграмма вариантов использования.

В приложении Б будет представлена структура сайта.

В приложении В будет представлена модель данных.

В приложении Г будет представлена функциональная модель.

В приложении Д будет представлена диаграмма деятельности.

В приложении Е будет представлена диаграмма классов.

В приложении Ж будет представлена диаграмма объектов.

В приложении З будет представлена диаграмма последовательности.

В приложении И будут представлены UX и UI прототипы.

В приложении К будут представлены тест-кейсы.

1. **Анализ задачи**
   1. **Постановка задачи**
      1. **Организационно-экономическая сущность задачи**

Наименование задачи**:** сайт Гродненского государственного политехнического колледжа.

Цель разработки**:** создание новой версии сайта Гродненского государственного политехнического колледжа.

Назначение**:** данный программный продукт предназначен для людей любого возраста с целью получения информации о колледже.

Периодичность использования**:** по мере необходимости.

Источники и способы получения данных**:** Сайт GGPK.by, сотрудники учреждения.

Обзор существующих аналогичных ПП**:** Рассмотрим сайт GGPK.by на базе которого будет разрабатываться новая версия. На этом сайте устарел дизайн, интерфейс, отсутствует авторизация для пользователя. Отсутствует среда для взаимодействия учащихся с преподавателями. Модуль «Расписание занятий» требует полного изменения.

* + 1. **Функциональные требования**

Описание перечня функций и задач, которые должен выполнять будущий ПП**:**

Гость:

1. Просмотр страниц сайта;
2. Просмотр информации о колледже;
3. Местоположение колледжа;
4. Переход в другие соц. сети такие как Instagram, Вконтакте;
5. Авторизация как пользователя;
6. Связь с администрацией;
7. График работы;
8. Блог приёмной комиссии;
9. Расписание занятий.

Пользователь: может всё тоже самое что и гость, а также появляется:

Учащийся:

1. Просмотр личных оценок;
2. Просмотр личной успеваемости;
3. Доступ к системе дистанционного обучения;
4. Получение учебного материала.

Преподаватель:

1. Выставление оценок учащимся;
2. Ведение ведомости у учащихся;
3. Доступ к системе дистанционного обучения;
4. Добавление учебной информации.

Администратор:

1. Доступ к админ. панели;
2. Редактирование страниц сайта;
3. Доступ к базам данных;
4. Доступ к аккаунтам пользователей;
5. Изменение расписания занятий;
6. Создание аккаунтов пользователей.
   * 1. **Описание процессов с входной, выходной и условно-постоянно информацией**

Описание исходной(входной) информации:

1. Логин пользователя;
2. Пароль пользователя.

Описание результатной (выходной) информации:

1. Учебный материал;
2. Данные об успеваемости.

Описание используемой условно-постоянной информации

1. Контакты;
2. Расписание занятий;
3. Информация о колледже;
4. Фотогалерея;
5. Карта сайта.
   * 1. **Нефункциональные (эксплуатационные) требования**

* Требования к применению:помогает узнать всю необходимую информацию об колледже для посетителей сайта, а также взаимодействовать преподавателям с учащимися;
* Требования к реализации:для реализации статических страниц и шаблонов должна использоваться система управления содержимым сайта WordPress;
* Требования к надёжности: Система может быть недоступна не более чем 24 часа в год. У администратора сайта должна быть возможность выгрузить и загрузить копию сайта;
* Требования к интерфейсу:При разработке сайта должны быть использованы преимущественного белого цвета с оттенками синего. Основные разделы сайта должны быть доступны с первой страницы. Грамотный пользовательский интерфейс. Сайт должен адаптироваться под компьютер, телефон и планшет.
  1. **Диаграмма вариантов использования**

В качестве актеров представлен гость, учащийся, преподаватель и администратор которые взаимодействует с системой.

Гость может:

* Просматривать страницы сайта;
* Переходить в другие соц. сети;
* Связаться с администрацией;
* Авторизоваться как пользователь;
* Просмотреть местоположение колледжа;
* Написать обращение.

Учащийся может:

* Всё тоже что и гость;
* Иметь доступ к системе дистанционного обучения.

Преподаватель может:

* Всё тоже что гость и учащийся;
* Иметь доступ к системе дистанционного обучения.

Администратор:

* Всё что и другие актеры;
* Иметь доступ к админ. панели;
* Иметь доступ к базе данных.

Разработанная диаграмма вариантов использования представлена в приложении А.

* 1. **Выбор стратегии разработки и модели жизненного цикла**

Для разработки веб-ресурса «сайт Гродненского государственного политехнического колледжа» 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** |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
| 1. | Являются ли требования к проекту легко определимыми и реализуемыми? | Да | Да | Да |  |  |  |
| 2. | Могут ли требования быть сформулированы в начале ЖЦ? | Да | Да | Да | Да |  |  |
| 3. | Часто ли будут изменяться требования на протяжении ЖЦ? |  |  |  |  | Да | Да |
| 4. | Нужно ли демонстрировать требования с целью их определения? | Нет | Нет |  | Нет |  |  |
| 5. | Требуется ли проверка концепции программного средства или системы? | Нет | Нет |  | Нет |  |  |
| 6. | Будут ли требования изменяться или уточняться с ростом сложности системы (программного средства) в ЖЦ? |  |  |  | Да | Да | Да |

Продолжение таблицы 1

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **№ критерия** | **Критерии категории требований** | **Каскадная** | **V-образная** | **RAD** | **Инкрементная** | **Быстрого прототипирования** | **Эволюционная** |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
| 7. | Нужно ли реализовать основные требования на ранних этапах разработки? |  |  | Да | Да | Да | Да |

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

Итог: На основе результатов заполнения табл. 3 подходящей является инкрементная 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** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
| 1. | Являются ли проблемы предметной области проекта новыми для большинства разработчиков? | Нет | Нет | Нет | Нет |  |  |
| 2. | Являются ли инструментальные средства, используемые в проекте, новыми для большинства разработчиков? |  |  | Нет | Нет | Нет |  |
|  |  |  |  |  |  |  |  |

Продолжение таблицы 2

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **№ критерия** | **Критерии категории требований** | **Каскадная** | **V-образная** | **RAD** | **Инкрементная** | **Быстрого прототипирования** | **Эволюционная** |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
| 3. | Изменяются ли роли участников проекта на протяжении ЖЦ? | Нет | Нет | Нет |  |  |  |
| 4. | Является ли структура процесса разработки более значимой для разработчиков, чем гибкость? | Да | Да |  | Да |  |  |
| 5. | Важна ли легкость распределения человеческих ресурсов проекта? |  |  |  |  | Нет | Нет |
| 6. | Приемлет ли команда разработчиков оценки, проверки, стадии разработки? | Да | Да |  | Да | Да | Да |

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

Итог: На основе результатов заполнения табл. 4 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** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
| 1. | Будет ли присутствие пользователей ограничено в ЖЦ разработки? | Да | Да |  | Да |  | Да |
| 2. | Будут ли пользователи оценивать текущее состояние программного продукта (системы) в процессе разработки? |  |  |  | Да | Да | Да |
| 3. | Будут ли пользователи вовлечены во все фазы ЖЦ разработки? | Нет | Нет |  | Нет |  | Нет |
| 4. | Будет ли заказчик отслеживать ход выполнения проекта? |  |  |  |  | Да | Да |

Вычисления: 2 за каскадную, 2 за V-образную, 0 за RAD, 3 за инкрементную, 2 за быстрого прототипирования и 4 за эволюционную.

Итог: На основе результатов заполнения табл. 5 подходящей является эволюционная модель.

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** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
| 1. | Разрабатывается ли в проекте продукт нового для организации направления? | Нет | Нет | Нет |  |  |  |
| 2. | Будет ли проект являться расширением существующей системы? | Да | Да | Да | Да |  |  |
| 3. | Будет ли проект крупно- или среднемасштабным? | Нет | Нет | Нет |  |  |  |
| 4. | Ожидается ли длительная эксплуатация продукта? | Да | Да |  | Да |  | Да |
| 5. | Необходим ли высокий уровень надежности продукта проекта? |  | Да |  | Да |  | Да |
| 6. | Предполагается ли эволюция продукта проекта в течение ЖЦ? |  |  |  | Да | Да | Да |
| 7. | Велика ли вероятность изменения системы (продукта) на этапе сопровождения? |  |  |  | Да | Да | Да |
| 8. | Является ли график сжатым? | Нет | Нет |  |  |  |  |
| 9. | Предполагается ли повторное использование компонентов? |  |  | Да | Да | Да | Да |

Вычисления: 5 за каскадную, 6 за V-образную, 4 за RAD, 6 за инкрементную, 4 за быстрого прототипирования и 6 за эволюционную.

Итог: На основе результатов заполнения табл. 6 подходящей является инкрементная, V-образная и эволюционная модели.

Общий итог: в итоге заполнения табл. 3 – 6 наиболее подходящей является инкрементная модель.

* 1. **Инструменты разработки**

Для создания новой версии сайта колледжа мы используем WordPress. WordPress (WP) ― это система управления содержимым сайта (CMS) с открытым исходным кодом. Самым популярным на данный момент способом создания сайта является именно CMS. CMS (Content Management System) ― это система создания и управления сайтом. WordPress обладает такими преимуществами как:

* WordPress обладает интуитивно понятным интерфейсом, что позволяет легко создавать и обновлять контент, даже если у пользователя нет глубоких технических знаний.
* Огромное сообщество разработчиков и пользователей гарантирует наличие множества плагинов, тем и документации, что позволяет находить решения для большинства задач.
* С помощью плагинов и тем WordPress можно адаптировать под любые нужды — от простого блога до сложного интернет-магазина.
* WordPress предоставляет отличные возможности для SEO, что важно для продвижения контента в поисковых системах.

1. **Проектирование**
   1. **Разработка системы меню и навигации по программе**

Система меню и навигации является важным компонентом пользовательского интерфейса, обеспечивающим удобный доступ к функциональности и разделам приложения. Ее главная задача — помочь пользователю быстро находить необходимую информацию и перемещаться между различными страницами. Для этого меню должно быть интуитивным, доступным и логичным.

При разработке такой системы необходимо учитывать адаптивный дизайн, чтобы меню корректно отображалось на любых экранах. Также важно документировать структуру и работу навигации, чтобы пользователи могли быстро освоиться. Хорошо продуманная система меню способствует повышению эффективности работы с приложением и улучшению общего пользовательского опыта.

Полную структуру сайта можно рассмотреть в приложении Б.

* 1. **Разработка UML- диаграмм**
     1. **Модель данных**

Модель данных — это логическая структура, описывающая, как данные хранятся, организованы и связаны в рамках информационной системы. Она определяет основные сущности, их атрибуты, взаимосвязи и ограничения, обеспечивая основу для обработки данных. Использование модели данных помогает эффективно управлять информацией, гарантировать целостность данных и упрощать работу с ними в процессе разработки и эксплуатации программного обеспечения.

Разработанная модель данных представлена в приложении В.

* + 1. **Функциональная модель**

Функциональная модель — это описание процессов и функций системы, показывающее, как она обрабатывает данные и взаимодействует с внешней средой. Она определяет, какие операции выполняются, их последовательность, входные и выходные данные, а также связь между различными функциями. Функциональная модель используется для анализа и проектирования системы, чтобы понять, как она должна работать и какие задачи выполнять для достижения поставленных целей.

В функциональной модели изображен процесс добавления новости на сайт.

С функциональной моделью можно ознакомиться в приложении Г.

* + 1. **Диаграмма деятельности**

Диаграмма деятельности — это графическое представление последовательности действий, потоков управления и данных в процессе или системе. Она показывает, как выполняются задачи, переходы между действиями и возможные параллельные процессы. Диаграмма деятельности помогает визуализировать логику работы системы, выявлять точки взаимодействия и оптимизировать процессы. Чаще всего используется при моделировании бизнес-процессов или проектировании программного обеспечения.

На диаграмме изображено добавление редактирование новостей и разделов на сайте администратором.

Диаграмма изображена в приложении Д.

* + 1. **Диаграмма классов**

Диаграмма классов – структурная диаграмма языка моделирования UML, демонстрирующая общую структуру иерархии классов системы, их коопераций, атрибутов (полей), методов, интерфейсов и взаимосвязей (отношений) между ними.

На диаграмме показаны используемые в программе классы.

С диаграммой можно ознакомиться в приложении Е.

* + 1. **Диаграмма объектов**

Диаграмма объектов — это графическое представление состояния системы в конкретный момент времени, показывающее объекты и их взаимосвязи. Она фокусируется на экземплярах классов, их атрибутах и связях, а не на общих структурах. Диаграмма объектов помогает понять, как данные взаимодействуют в определенной ситуации или этапе работы системы, и часто используется для уточнения и проверки моделей данных или диаграмм классов.

На диаграмме показано взаимодействие администратора с меню добавления новостей.

Диаграмма объектов показана в приложении Ж.

* + 1. **Диаграмма последовательности**

Диаграмма последовательности используется для визуализации взаимодействия между объектами в системе. Она показывает порядок сообщений, которыми объекты обмениваются друг с другом во времени и их жизненный цикл. Это самый простой и удобный инструмент для демонстрации всех интеграций и взаимодействий в рамках проектируемого бизнес-процесса.

На диаграмме показана последовательность действий для написания обращения на сайте.

Диаграмма представлена в приложении З.

* 1. **Разработка пользовательского интерфейса**

Одним из важнейших элементов проектирования является описание внешнего интерфейса разрабатываемого программного продукта.

Для обеспечения эффективной работы пользователя нужно создать целостное приложение, охватывающее конкретную предметную область, где все компоненты будут сгруппированы по их функциональному назначению. При этом необходимо обеспечить удобный графический интерфейс. Приложение должно позволять пользователю решать задачи с минимальными усилиями по сравнению с работой с разрозненными объектами.

Прототип – это модель пользовательского интерфейса. В сущности, это «черновик», созданный на основе представления разработчика о потребностях пользователя.

В создании интерфейса был задействован шрифт Inter. Для текста были использованы черные, серые и белые цвета. Для создания цветовой стилистики основными цветами были выбраны: белый, синий, черный и серый. Логотип сайта был взят со старой версии сайта Гродненского государственного политехнического колледжа.

Прототипы UX и UI представлены в приложении И.

Со всеми разработанными UX-прототипами можно ознакомиться по ссылке:

https://www.figma.com/design/CIZI5wsusM9gJl3PoVxPrc/UX-prototype?node-id=0-1&p=f&t=S2m6bG0HUFNp6gA0-0

Со всеми разработанными UI-прототипами можно ознакомиться по ссылке:

https://www.figma.com/design/AGMlGhwCjwzKDq7STjvkX3/UI-prototype?t=S2m6bG0HUFNp6gA0-0

1. **Реализация**
   1. **Руководство программиста**

Программный продукт разработан с использованием системы управления содержимым сайта WordPress. Этот ресурс представляет собой удобный и простой в использовании инструмент, позволяющий создавать сайты на основе готовых шаблонов тем и различных плагинов. Для начала работы необходимо установить OpenServer для загрузки на него сайта. В нем необходимо выбрать нужную версию PHP и MySQL. Далее устанавливаем на OpenServer phpMyAdmin в котором будут базы данных сайта. Устанавливаем WordPress на OpenServer и настраиваем. Создаём в phpMyAdmin новую базу данных для сайта.

* + 1. **Создание сайта**

Для того, чтобы создать свой интернет-ресурс в WordPress, необходимо сначала зарегистрироваться. Данные для входа на сайт администратора:

E-mail: maks.kozeyko@gmail.com;

Пароль: admin.

После авторизации нам доступна админ-панель.

Админ-панель сайта показана на рисунке 1.

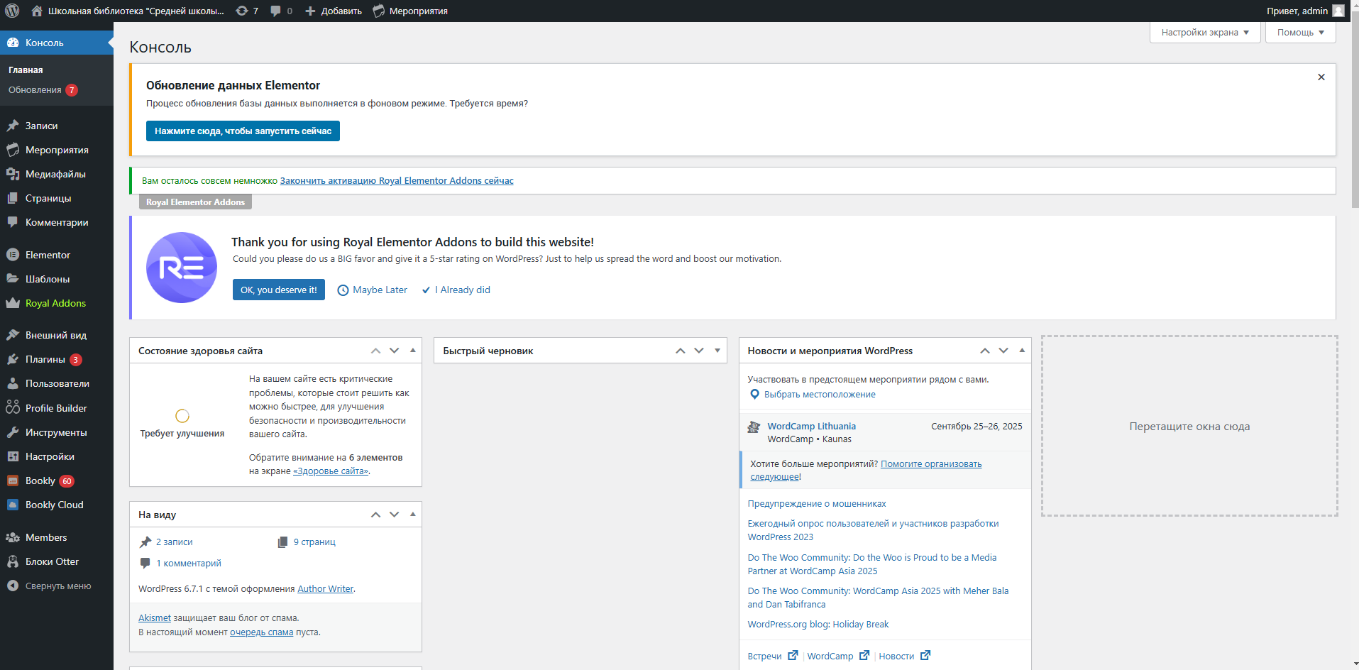


Рисунок 1 – Админ. панель

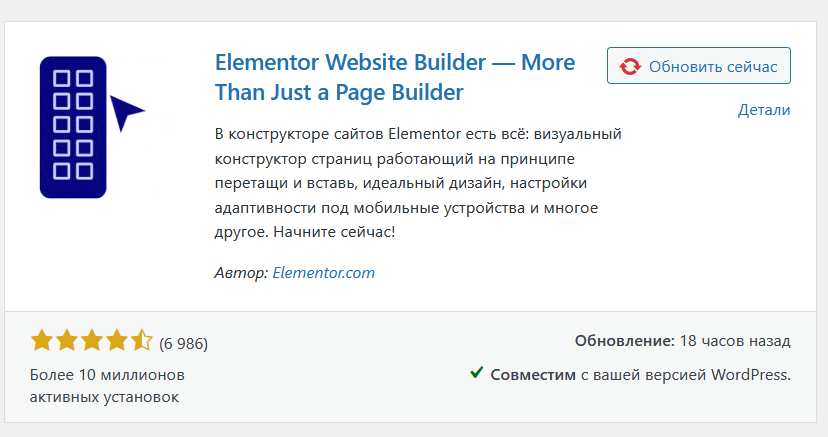
* + 1. **Настройка функциональности и дизайна**

За настройку функциональности отвечает консоль.

Здесь находится меню которое содержит следующие элементы:

* Записи – раздел в котором можно создавать и редактировать записи;
* Медиафайлы – раздел в котором можно загружать различные изображения, видео и т.д. для отображения их на сайте;
* Страницы – раздел для добавления и редактирования страниц сайта;
* Комментарии – раздел для просмотра написанных комментариев сайте и взаимодействия с ними;
* Шаблоны – раздел в котором можно сохранять и использовать шаблоны для тем и страниц;
* Внешний вид – раздел в котором можно выбирать темы сайта, паттерны, виджеты, создавать и редактировать меню;
* Плагины – раздел для добавления и управления плагинами;
* Пользователи – раздел для добавления и управления пользователями на сайте;
* Инструменты – раздел для работы с различными инструментами, импортами и экспортами.
* Настройки – раздел с настройками управления сайтом.
  + 1. **Редактирование страниц**

Для удобства работы со страницами необходимо установить плагин «Elementor» (рисунок 2).

Рисунок 2 – Плагин «Elementor»

Для создания страницы необходимо в меню найти кнопку «Страницы»,

далее «Создать страницу». Далее нажимаем кнопку «Редактировать в Elementor» для перехода в конструктор.

Панель редактирования состоит из таких кнопок как:

* Добавить элемент – для добавления элементов на страницу;
* Настройки сайта – для открытия меню глобальной настройки сайта;
* Структура – для открытия и просмотра структуры элементов на странице;
* Недавние – для просмотра недавно измененных страниц;
* Настройки страницы – для просмотра настроек страницы;
* Режим просмотра – для редактирования сайта для таких устройств как компьютер, телефон, планшет;
* Просмотреть изменения – для открытия версии предварительного просмотра страницы;
* Опубликовать – для сохранения изменений на странице.

На рисунке 3 представлено меню редактирования страницы.



Рисунок 3 – меню редактирования страницы

* + 1. **Настройка header и footer**

Header сайта будет содержать логотип сайта, название сайта, кнопку «Темный режим», кнопку «Выбор языка», кнопку «Режим для слабовидящих», кнопку «Полезные телефоны», кнопку «Написать обращение», кнопку «Профиль пользователя» и меню с разделами сайта. Header сайта представлен на рисунке 4.



Рисунок 4 – Header сайта

Footer сайта будет содержать карту с меткой колледжа, общее описание сайта и кнопки для перехода в соц. сети колледжа.

Просмотреть footer сайта можно на рисунке 5.

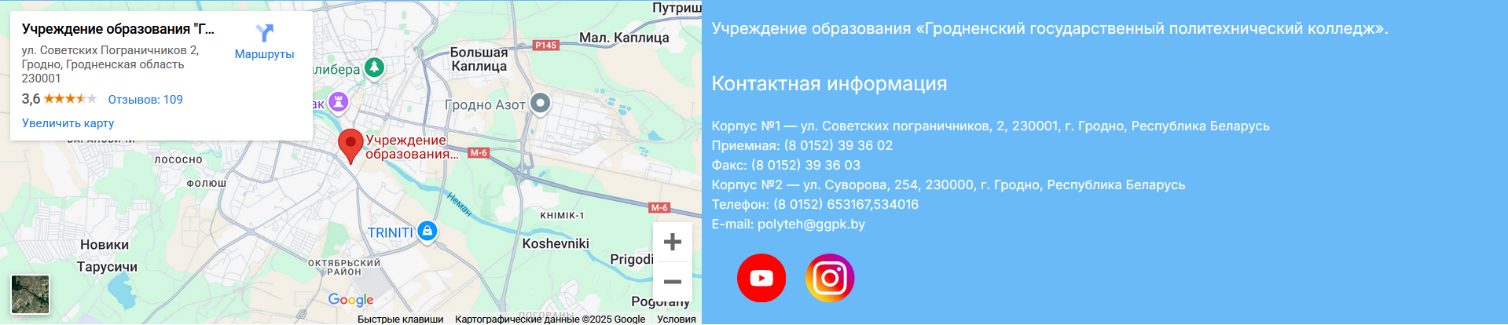


Рисунок 5 – Footer сайта

Для добавления header и footer сайта необходимо в консоли выбрать «Шаблоны», далее перейти в «Theme Builder». Ищем пункт «Header» и нажимаем кнопку добавить. Такой же алгоритм повторяем для footer. Редактирование header и footer такое же как редактирование страниц.

1. **Тестирование**
   1. **Тесты на использование**

При разработке программы возникали различные ошибки и недоработки, которые исправлялись в ходе реализации. После завершения реализации программы было проведено полное тестирование программы. Тестирование должно гарантировать стабильную и бесперебойную работу программы.

Разработанные тест-кейсы и статус их выполнения представлены в приложении К.

Расписание работ над проектом представлено в таблице 5.

Таблица 5 – Расписание работы над проектом

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Дата | Деятельность | Продолжительность, ч |
| Козейко Максим | 06.12.2024 | Разработка тестов | 2 |
| Козейко Максим | 06.12.2024 | Тестирование программы | 2 |
| Козейко Максим | 08.11.2024 | Составление отчетов о найденных дефектах | 1 |
| Козейко Максим | 09.11.2024 | Исправление найденных ошибок | 3 |
| Козейко Максим | 11.11.2024 | Проведение регрессионного тестирования | 2 |
| Козейко Максим | 11.11.2024 | Составление отчета о результатах тестирования | 2 |

* 1. **Отчёт о результатах тестирования**

Элементы программы были проверены, и было установлено, что все они работают правильно и выполняют задачи, указанные в процедурах.

Статистика по всем дефектам представлена в таблице 6.

Таблица 6 – Статистика по всем дефектам

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | Важность | | | |
| Статус | Количество | Низкая | Средняя | Высокая | Критическая |
| Найдено | 0 | 0 | 0 | 0 | 0 |
| Исправлено | 0 | 0 | 0 | 0 | 0 |
| Проверено | 0 | 0 | 0 | 0 | 0 |
| Открыто заново | 0 | 0 | 0 | 0 | 0 |
| Отклонено | 0 | 0 | 0 | 0 | 0 |

1. **Руководство пользователя**

Для того, чтобы открыть сайт необходимо перейти по ссылке.

После открытия сайта загружается главная страница, представленная на рисунке 6.

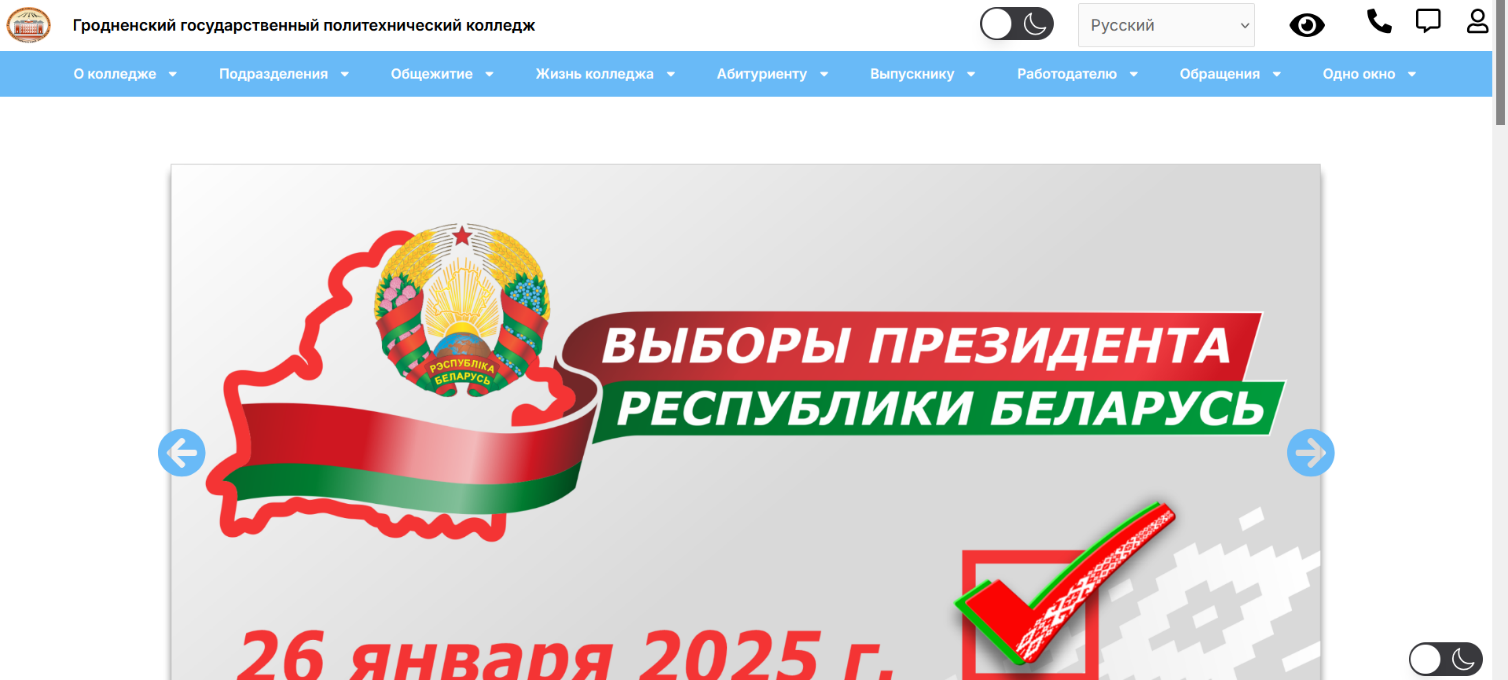


Рисунок 6 – Главная страница

Переход по страницам можно осуществлять по горизонтальному меню.

Страница «Общая информация» включает в себя адрес колледжа, время работы колледжа и справочник телефонов колледжа. На рисунке 7 представлена страница «Общая информация».

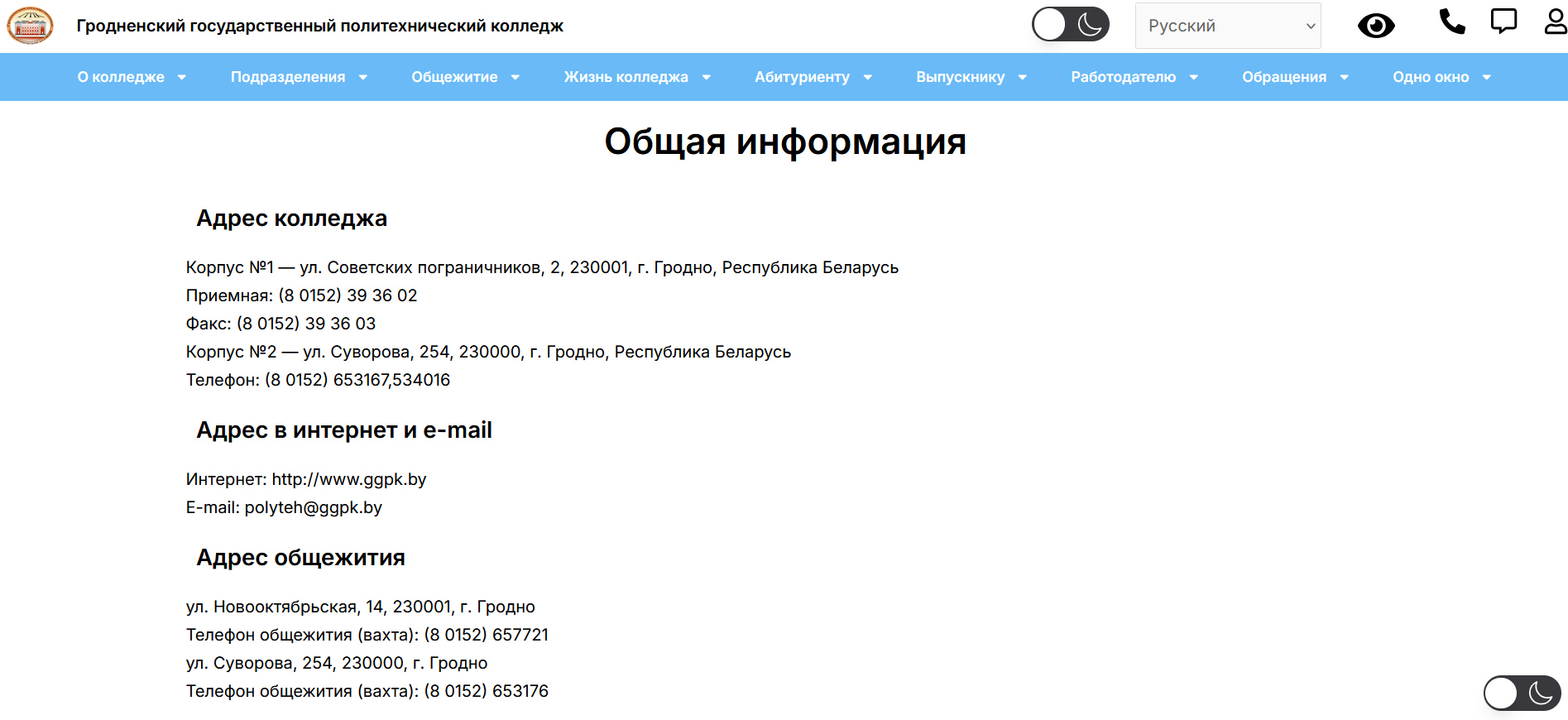
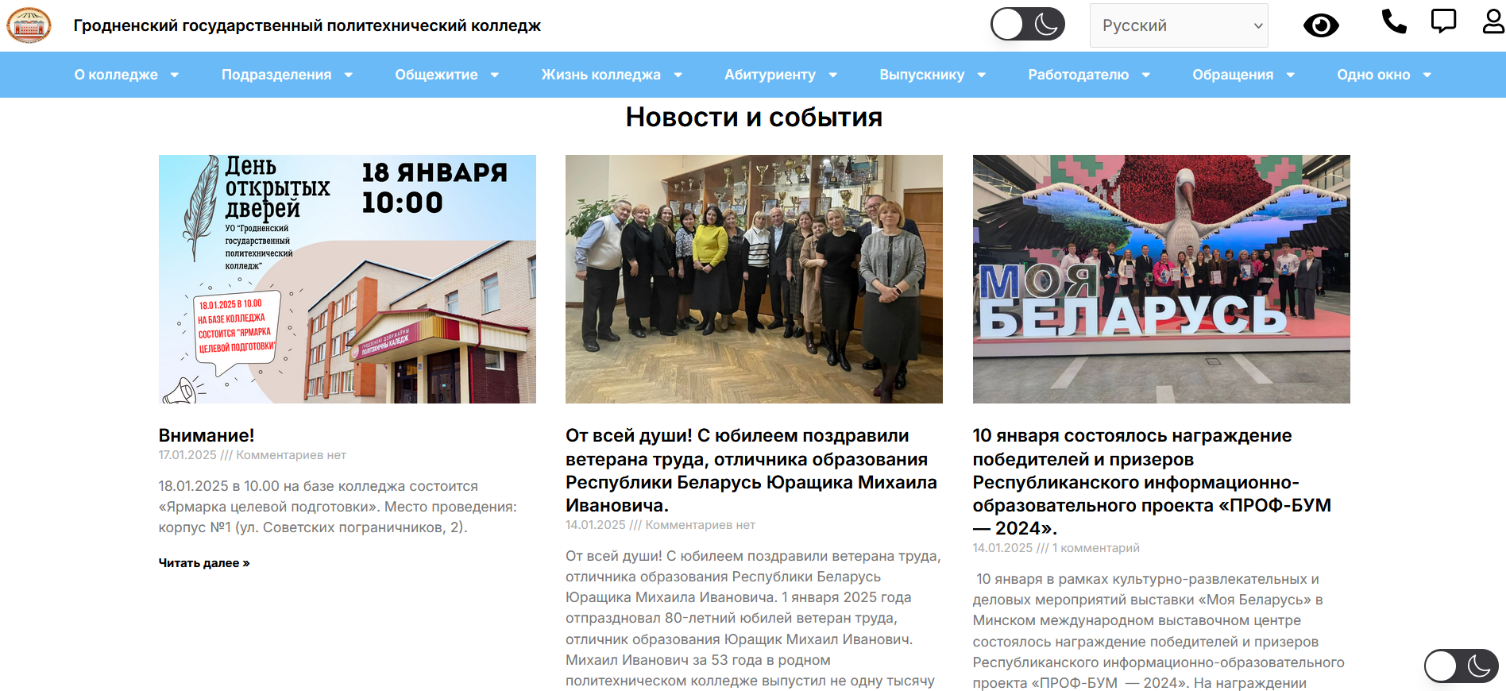


Рисунок 7 – Страница «Общая информация»

На странице «Новости и события» представлен архив новостей сайта. Здесь можно перейти на страницу каждой новости. На рисунке 8 представлена страница «Новости и события».

Рисунок 8 – Страница «Новости и события»

На странице новости можно подробно ознакомиться с новостью. Здесь можно перейти к похожим записям (рисунок 9) и оставить комментарии (рисунок 10).

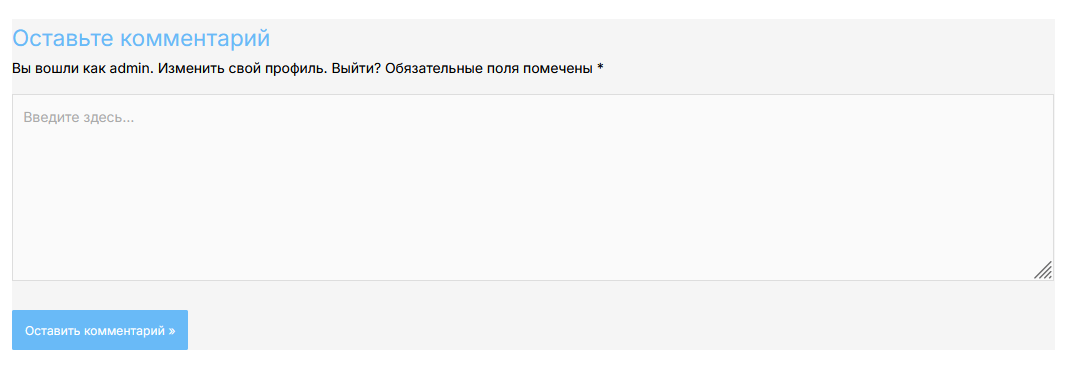
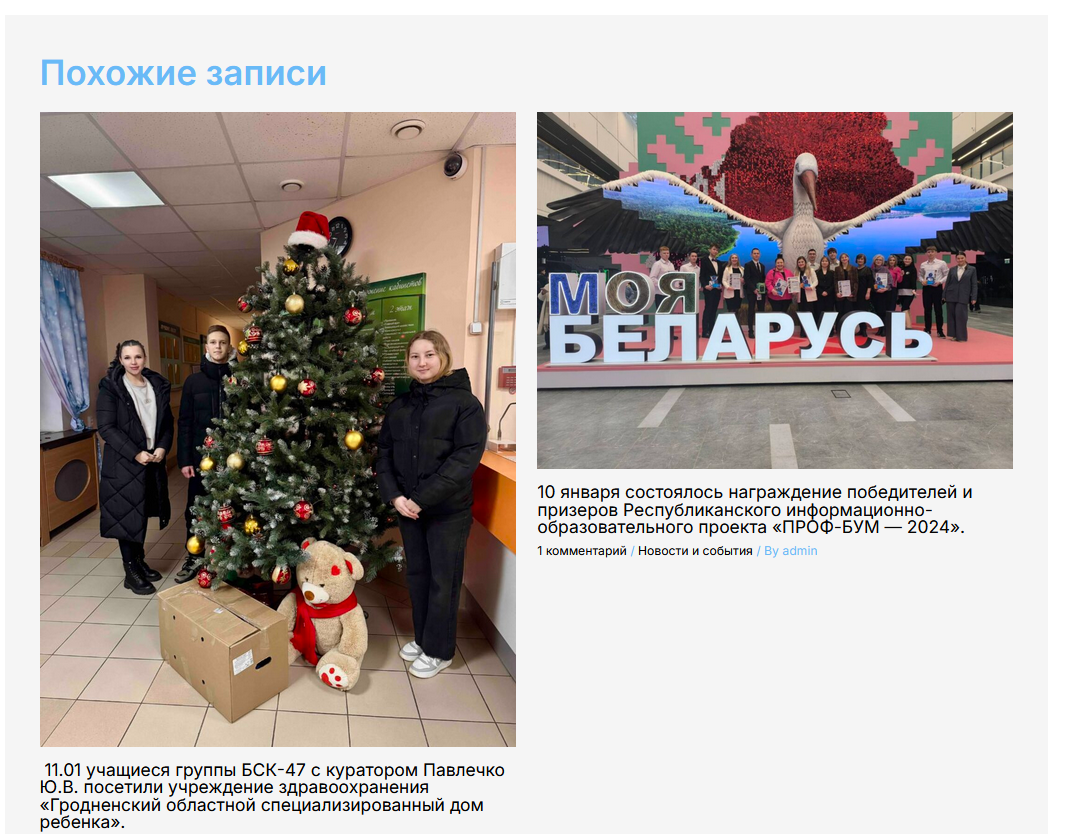


Рисунок 9 – Похожие записи

Рисунок 10 – Комментарии

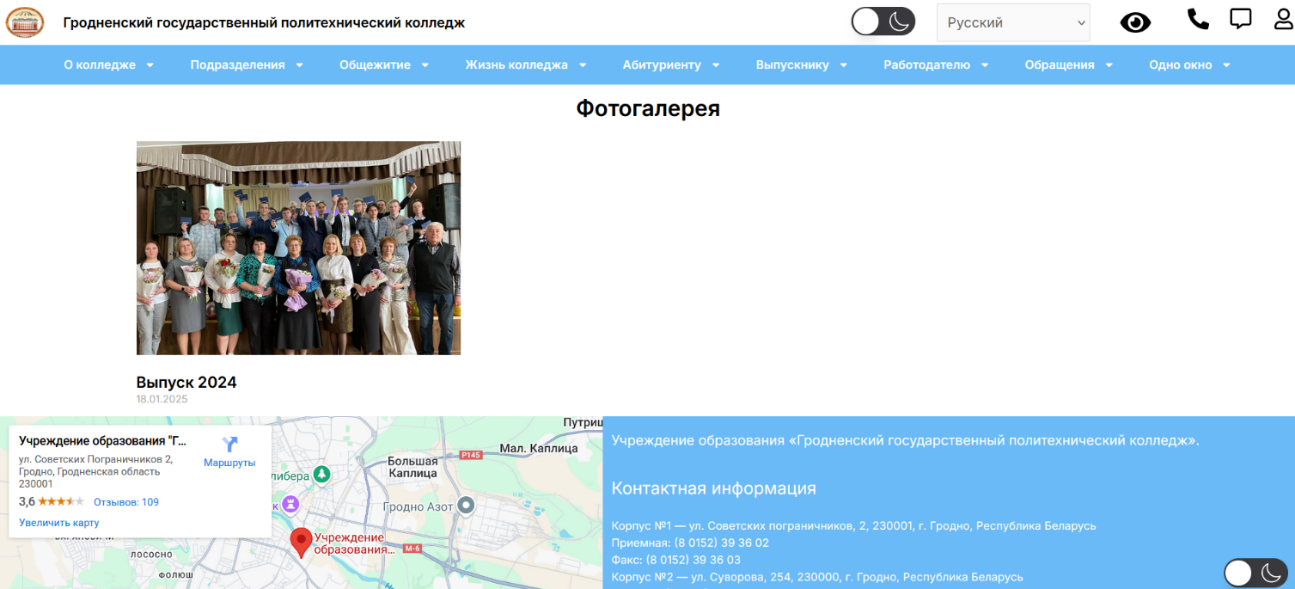
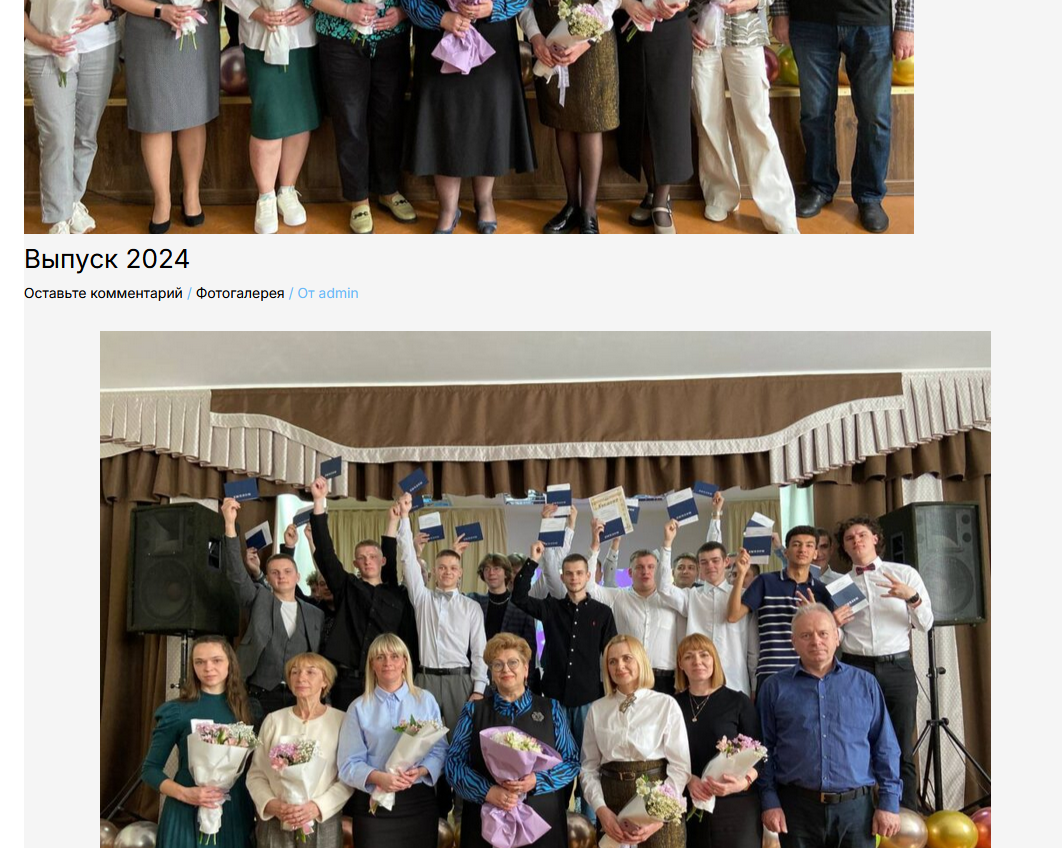
Из главной страницы можно перейти на страницу «Фотогалерея». На этой странице содержится архив всех фотографий (рисунок 11). Здесь можно выбрать альбом.

Рисунок 11 – Страница «Фотогалерея»

На странице «Альбом» можно просмотреть все фотографии альбома (рисунок 12) и оставлять комментарии.

Рисунок 12 – Страница «Альбом»

**Заключение**

Целью данного проекта является создание новой версии сайта «Гродненский государственный политехнический колледж», который позволит узнавать информацию о колледже, время работы, расписание занятий и телефоны связи.

В ходе реализации поставленной задачи были закреплены знания по использованию системы управления содержимым сайта WordPress, а также курс предмета «Технология разработки программного обеспечения».

В ходе тестирования все исключительные ситуации были обработаны.

Проект работает без сбоев и ошибок. В поставленной задаче был реализован простой и понятный пользовательский интерфейс.

Исходя из этого, можно сделать вывод, что программа реализована успешно.

**Список использованных источников**

1. Как создать сайт на WordPress [Электронный ресурс]. – Режим доступа: https://texterra.ru/blog/kak-sozdat-sayt-na-wordpress-polnoe-rukovodstvo-dlya-novichkov.html – Дата доступа: 15.12.2024.
2. Что такое WordPress: о программе и работе с ней [Электронный ресурс]. – Режим доступа: https://help.reg.ru/support/hosting/cms/chto-takoye-wordpress-o-programme-i-rabote-s-ney – Дата доступа: 16.12.2024.