

Введение

На учебной практике была поставлена задача разработать web-ресурс на тему: «Спортивная жизнь УО «Гродненский государственный политехнический колледж».

Цель проекта – разработка web-ресурса, который познакомит преподавателей и обучающихся с новостями спортивной жизни Гродненского государственного политехнического колледжа, позволит преподавателям хранить всю необходимую документацию для организации спортивных мероприятий, а учащимся – изучать материал и узнавать о спортивной жизни колледжа. Web-ресурс будет иметь удобный и интуитивно понятный интерфейс, который позволит преподавателям и учащимся легко находить нужную информацию. Ресурс будет поддерживать различные форматы файлов, включая текстовые документы, фотографии и таблицы.

Далее приведено краткое описание разделов пояснительной записки.

Первый раздел носит название «Анализ задачи». В нем можно ознакомиться с постановкой задачи, которая включает в себя: исследование предметной области поставленной задачи, инструменты разработки (будет рассмотрена среда, в которой создается данный проект), диаграмму Ганта и выбор модели жизненного цикла программного обеспечения. Также в этом разделе можно узнать о том, как данная задача решается в настоящее время. Все входные и выходные данные тоже будут описаны в первом разделе.

В разделе «Проектирование задачи» будут рассмотрены основные аспекты разработки интернет-ресурса. 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.

Четвёртый раздел – «Тестирование». В нем будет описано полное и функциональное тестирование данной программы. 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 приведён список используемых при разработке источников.

					УП ТРПО 2-40 01 01.35.40.02.24 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		4

В приложениях к пояснительной записке будут приведены UX и UI проектирование сайта, диаграммы и тест-кейсы.

					УП ТРПО 2-40 01 01.35.40.02.24 ПЗ	Лист
						5
Изм.	Лист	№докум.	Подпись	Дата		

1 Анализ задачи

1.1 Постановка задачи

1.1.1 Организационно-экономическая сущность задачи

На проектирование была поставлена задача разработать веб-сайт, в котором пользователь сможет посмотреть материал по изучению C#.

Периодичность программного продукта:

Данный сайт разрабатывается для всех пользователей без ограничений по возрасту и места проживания. Данный программный продукт будет использоваться довольно часто, т.к. есть много желающих хотели бы больше узнать о таком языке программирования как C#.

Предназначение программного продукта:

Данный сайт будет иметь функции, такие как просмотр материала, упражнения и задания после освоения каждой темы для самопроверки с системой оценивания. Также, на данном продукте можно будет найти определённые темы.

Основным пользователем данного интернет ресурса будет любой желающий научиться основам C#.

1.1.2 Функциональные требования

Требования незарегистрированного пользователя:

- просмотр теоретического материала;
- выполнение упражнений для самопроверки;
- выполнение теста для самопроверки.

Требования администратора:

- просмотр теоретического материала;
- выполнение упражнений для самопроверки;
- выполнения теста для самопроверки;
- добавление/удаление/изменение теоретического материала;
- добавление/удаление/изменение упражнений;
- добавление/удаление/изменение тестов.

1.1.3 Описание процессов с входной, выходной и условно-постоянной информацией

Входная информация:

- данные учащегося.

Выходная информация:

- результаты тестов.

					УП ТРПО 2-40 01 01.35.40.02.24 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		6

Условно-постоянная информация:

- ФИО администратора;
- телефон администратора;
- список теоретического материала(в каталоге);
- список заданий для самопроверки(в каталоге);
- список тестов для самопроверки(в каталоге).

1.1.4 Эксплуатационные требования

Требования к применению: Данный программный продукт помогает находить определённый теоретический материал по С#.

Требования к реализации: Данный программный продукт нужно реализовать в виде веб-сайта. Для создания данного программного продукта нужно будет использовать конструктор сайтов WordPress.

Требования к надёжности: На данном веб-сайте осуществляется защита пользовательских данных от взлома.

Требования к хостингу:

- Круглосуточный мониторинг состояния сайта и своевременное устранение ошибок;
- Защита от взлома и спама;
- Все данные проходят через процедуру ежедневного резервного копирования. Копии хранятся в течение 20 дней.

Требования к интерфейсу: Данный программный продукт должен иметь интерфейс, соответствующий обучающей теме, удобную навигацию, систему классификации пользователей(гость/администратор).

1.2 Диаграмма вариантов использования

Диаграмма вариантов использования содержит в себе информацию о действиях каждого вида пользователей(гость/зарегистрированный пользователь/администратор).

Для данного проекта было выделено только 2 актёра – гость и администратор.

Диаграмма вариантов использования изображена на рисунке 1.1.

					УП ТРПО 2-40 01 01.35.40.02.24 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		7



Рисунок 1.1 – Диаграмма вариантов использования

1.3 Инструменты разработки

Данное электронное средство обучения разработана в конструкторе сайтов WordPress 6.6.2.

WordPress — свободно распространяемая система управления содержимым сайта с открытым исходным кодом; написана на PHP; сервер базы данных — MySQL; выпущена под лицензией GNU GPL версии 2. Сфера применения — от блогов до достаточно сложных новостных ресурсов. Встроенная система «тем» и «плагинов» вместе с удачной архитектурой позволяет конструировать проекты широкой функциональной сложности.

Также, в ней будет удобнее создавать отдельные функции, вместо того, чтобы абсолютно каждое действие нужно было бы прописывать на нескольких языках программирования сразу.

Для работы с WordPress понадобилась среда для веб-разработки Open Server.

Open Server - мощная портативная программная среда, созданная специально для веб-разработчиков, включающая в себя тщательно подобранный набор серверного программного обеспечения (Apache, Nginx, Sendmail, Ghostscript, PHPMyAdmin, MariaDB, Memcached, MongoDB, MySQL, PostgreSQL, Redis, различные PHP модули и дополнительные расширения и пр.), а также удобную

управляющую утилиту, которая обладает мощными возможностями по администрированию и настройке всех доступных компонентов.

Кроме того, включен пакет программ для начинающих веб-разработчиков, как например Opera, Sublime, WinSCP, Simplewall, Jcpicker, Bright Master.

Данный инструмент широко используется с целью разработки, отладки и тестирования веб-проектов, а также для предоставления веб-сервисов в локальных сетях.

Интерфейс создавался через онлайн-сервис разработки интерфейсов figma. В figma наиболее удобно создать макеты сайта как для десктопного приложения, так и для планшета и телефона. Также, в figma можно проработать некоторые функции сайта, такие как гиперссылки. UX- и UI-прототипы изображены в figma.

Для оформления документации понадобилось приложение Microsoft Word 2013 года.

1.4 Выбор модели жизненного цикла программного обеспечения

Рассмотрим критерии категорий требований, чтобы определить МЖЦПО.

Таблица 1.1 - 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 определяемыми и реализуемыми?	Да	Да	Да	Нет	Нет	Нет
2.	Могут ли требования быть сформулированы в начале ЖЦ?	Да	Да	Да	Да	Нет	Нет
Продолжение таблицы 1.1							
1	2	3	4	5	6	7	8
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

Каскадная – 4, V-образная – 4, RAD – 4, Инкрементная – 5, Быстрого прототипирования – 3, Эволюционная – 3.

Таблица 1.2 - 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	Да	Да
2.	Являются ли инструментальные средства, используемые в проекте, новыми для большинства разработчиков?	Да	Да	Нет	Нет	Нет	Да
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

Продолжение таблицы 1.2

1	2	3	4	5	6	7	8
6.	Приемлет ли команда разработчиков оценки, проверки, стадии разработки?	Да	Да	Нет	Да	Да	Да

Каскадная – 5, V-образная – 5, RAD – 4, Инкрементная – 5, Быстрого прототипирования – 2, Эволюционная – 1.

Таблица 1.3 - 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?	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да	<u>Нет</u>	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?	<u>Нет</u>	<u>Нет</u>	Да	Нет	Да	Нет
4.	Будет ли заказчик отслеживать ход выполнения проекта?	Нет	Нет	Да	Нет	Да	Да

Каскадная – 2, V-образная – 2, RAD – 3, Инкрементная – 3, Быстрого прототипирования – 4, Эволюционная – 2.

Таблица 1.4 - 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	Да	Да	Да

Продолжение таблицы 1.4

1	2	3	4	5	6	7	8
2.	Будет ли проект являться расширением существующей системы?	Да	Да	Да	Да	<u>Нет</u>	<u>Нет</u>
3.	Будет ли проект крупно- или среднемасштабным?	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да	Да	Да

4.	Ожидается ли длительная эксплуатация продукта?	<u>Да</u>	<u>Да</u>	Нет	<u>Да</u>	Нет	<u>Да</u>
5.	Необходим ли высокий уровень надежности продукта проекта?	Нет	<u>Да</u>	Нет	<u>Да</u>	Нет	<u>Да</u>
6.	Предполагается ли эволюция продукта проекта в течение ЖЦ?	Нет	Нет	Нет	<u>Да</u>	<u>Да</u>	<u>Да</u>
7.	Велика ли вероятность изменения системы (продукта) на этапе сопровождения?	Нет	Нет	Нет	<u>Да</u>	<u>Да</u>	<u>Да</u>
8.	Является ли график сжатым?	Нет	Нет	<u>Да</u>	<u>Да</u>	<u>Да</u>	<u>Да</u>
9.	Предполагается ли повторное использование компонентов?	Нет	Нет	<u>Да</u>	<u>Да</u>	<u>Да</u>	<u>Да</u>
10.	Являются ли достаточными ресурсы (время, деньги, инструменты, персонал)?	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да	Да

Каскадная – 3, V-образная – 4, RAD – 4, Инкрементная – 8, Быстрого прототипирования – 6, Эволюционная – 8.

Итог:

1. Инкрементная – 21 б.
2. V-образная, RAD и Быстрого прототипирования – 15 б.
3. Каскадная и Эволюционная – 14 б.

Т.е., больше всего, для создания данного ПП, понадобится Инкрементная модель проектирования.

1.5 Диаграмма Ганта

На рисунке 1.2 изображен план поэтапной разработки данного проекта – Диаграмма Ганта:

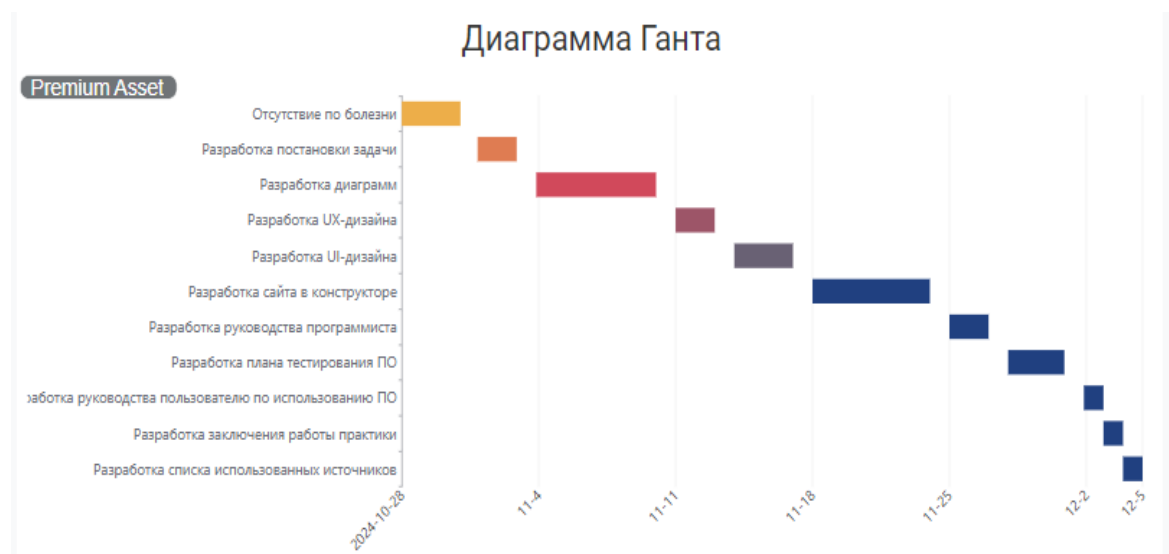


Рисунок 1.2 – Диаграмма Ганта

2 Проектирование задачи

2.1 Разработка пользовательского интерфейса

2.1.1 Структура сайта

Сайт имеет довольно простую структуру: Главная страница и страницы с теоретическим материалом, упражнениями и тестом.

Структура сайта изображена на рисунке 2.1.

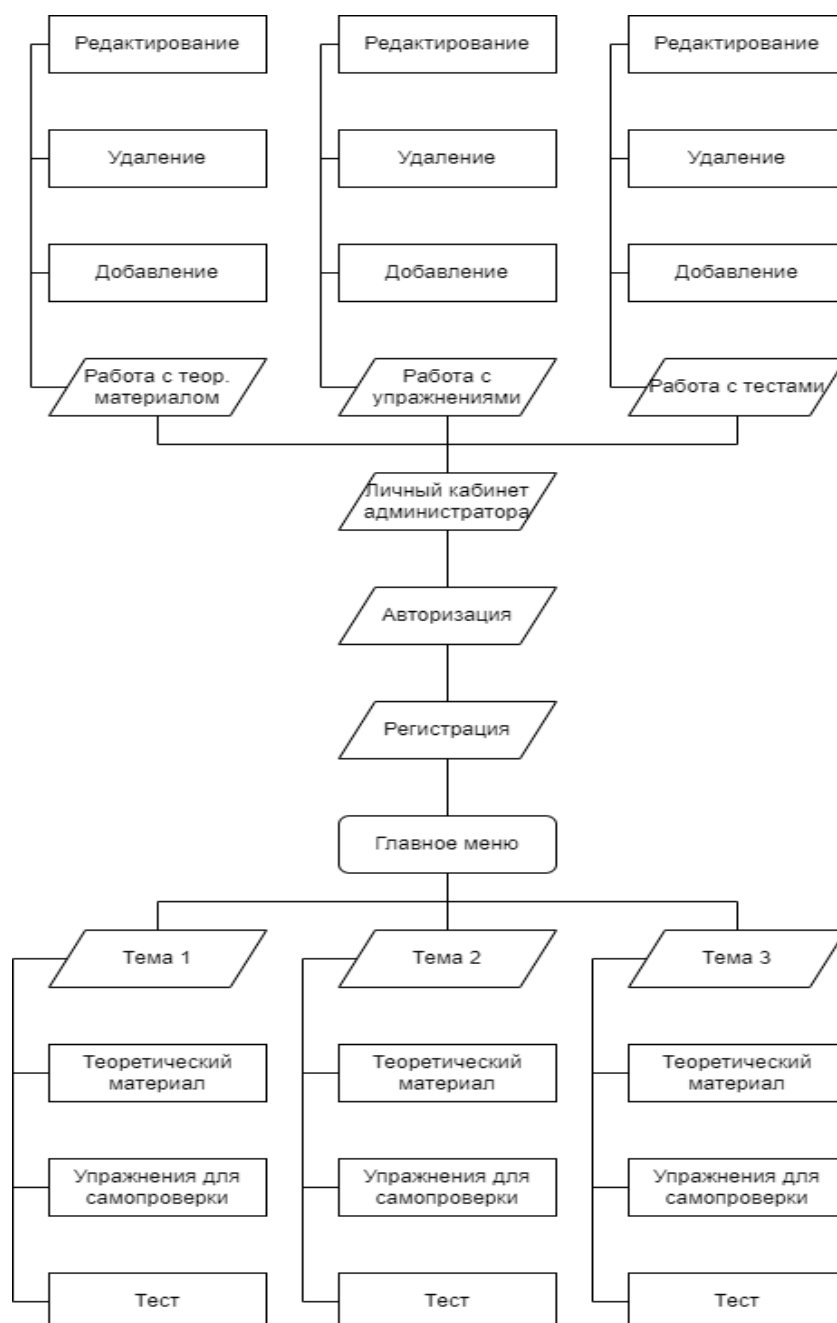


Рисунок 2.1 – Структура сайта

2.1.2 UX-прототипы пользовательского интерфейса

UX-прототипы пользовательского интерфейса сайта изображены на рисунке 2.2.

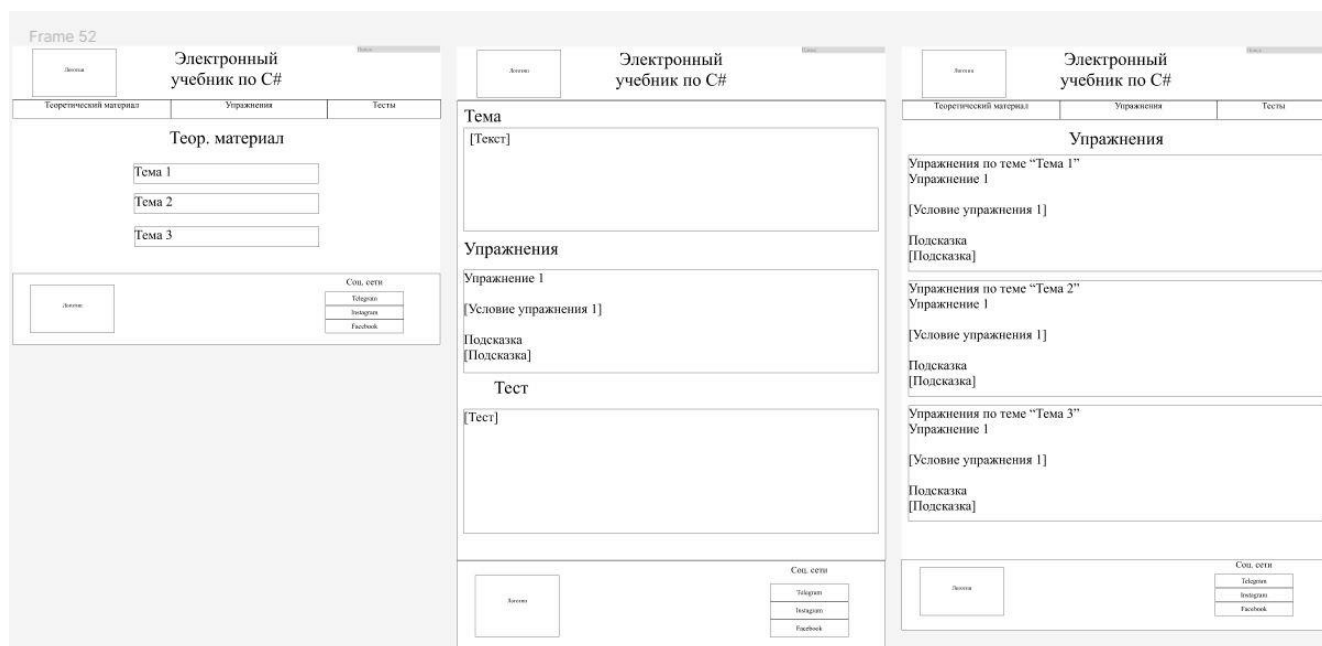


Рисунок 2.2 – UX-прототипы

2.1.3 UI-прототипы пользовательского интерфейса

UI-прототипы пользовательского интерфейса сайта изображены на рисунке 2.3.

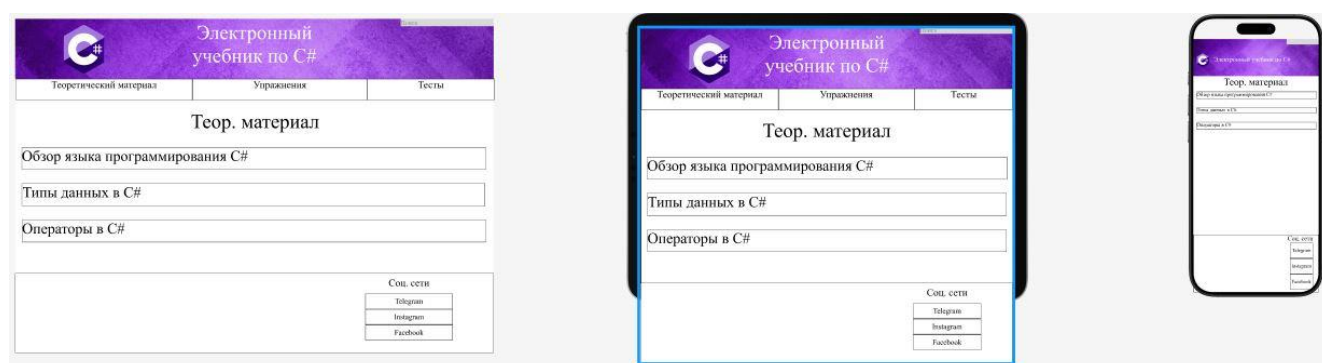


Рисунок 2.3 – UI-прототипы интерфейса сайта

2.2 Разработка UML-диаграмм

2.2.1 Модель данных

Модель данных отображает связи между компонентами сайта и администратором.

Модель данных изображена на рисунке 2.4.

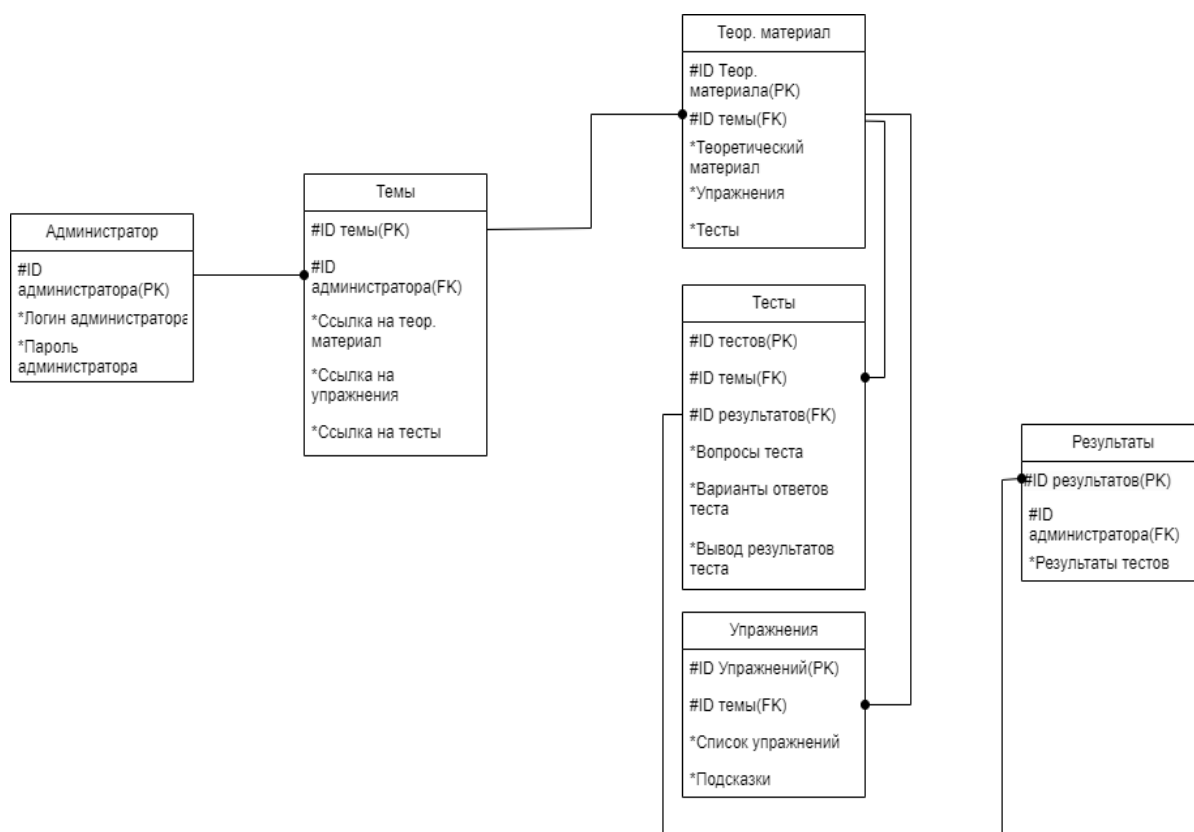


Рисунок 2.4 – Модель данных

2.2.2 Диаграмма последовательности

Диаграмма последовательности отображает последовательность действий для публикации администратором нового теоретического материала.

Диаграмма последовательности изображена на рисунке 2.5.

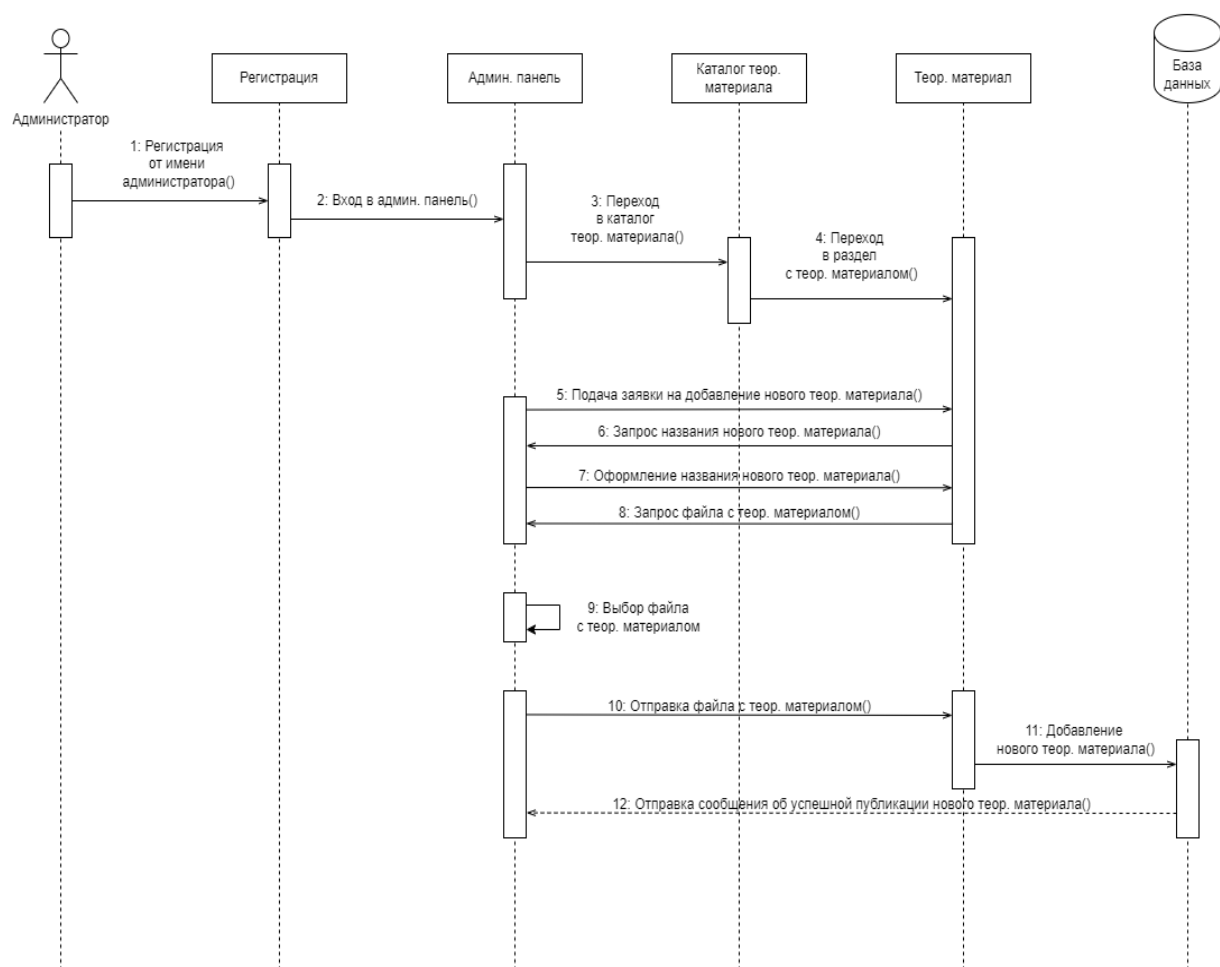


Рисунок 2.5 – Диаграмма последовательности

2.2.3 Диаграмма деятельности

Диаграмма деятельности отображает основные функции пользователя. Диаграмма деятельности изображена на рисунке 2.6.

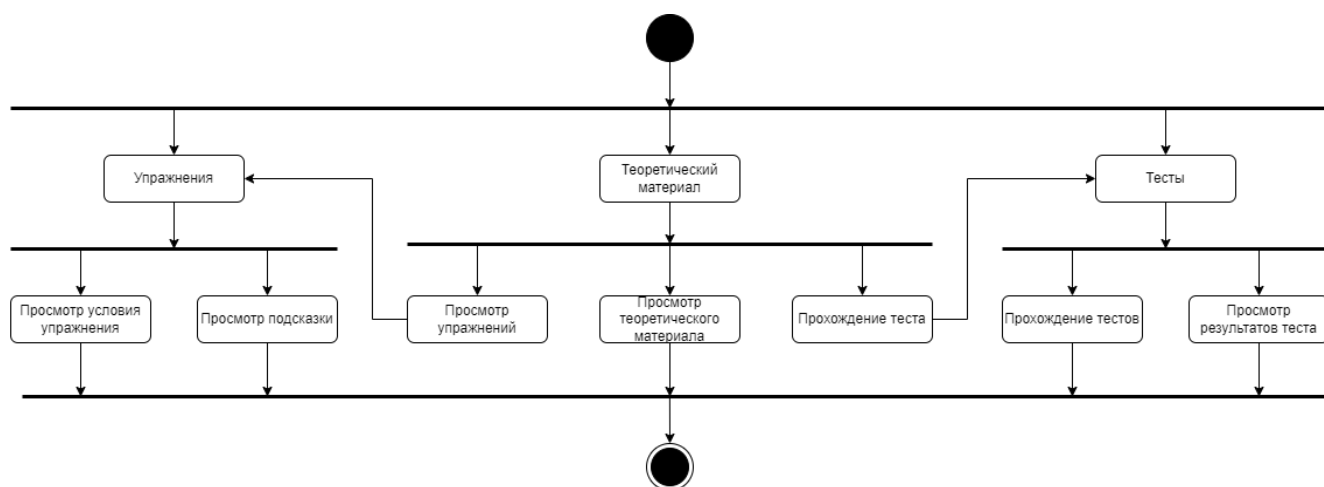


Рисунок 2.6 – Диаграмма деятельности

3 Реализация

3.1 Руководство программиста

3.1.1 Создание сайта

Данное электронное средство обучения разрабатывалось в конструкторе сайтов WordPress.

Для того, чтобы создать страницу сайта, нужно зайти во вкладку «Страницы» и нажать на кнопку «Добавить страницу», после чего создаётся страничка сайта, которую мы должны настраивать.

Добавление страницы изображено на рисунке 3.1.

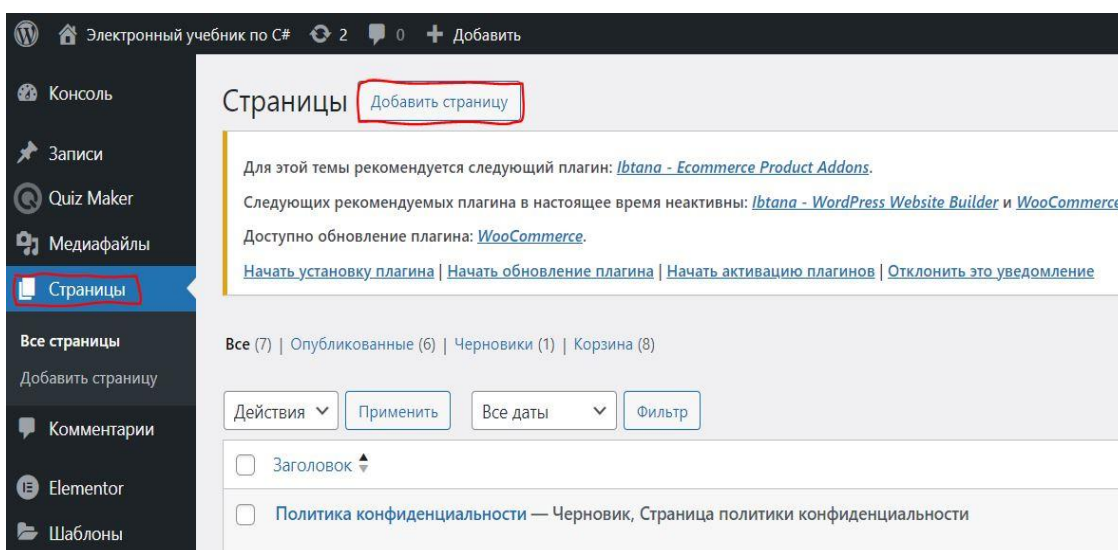


Рисунок 3.1 – Добавление страницы

Всегда при создании сайта добавляется большой заголовок, который несёт в себе само название страницы.

Далее, нужно добавлять отдельные блоки, которые будут нужны.

Для создания сайта были применены следующие блоки:

- заголовок – для обозначения названия темы/подтемы;
- абзац – для записи теоретического материала;
- код – для более чёткой записи кода;
- разделитель – для более удобной навигации в разделах теор. материала, упражнений и тестов;
- цитата – для обозначения основных понятий в разделе теор. материала;
- таблица – для изображения таблиц в теоретическом материале;
- Quiz Maker – для вставки теста, созданного в плагине Quiz Maker;
- кнопка – для создания кнопок перехода между страницами.

3.1.2 Создание внешнего вида сайта

Для придания внешнего вида сайта была применена тема VW Job Board.

Изначально, этой темы не будет и надо будет скачивать эту тему вручную.

Сначала, нужно войти во вкладку «Внешний вид» и нажать на кнопку «Добавить новую тему».

Добавление новой темы изображено на рисунке 3.2.

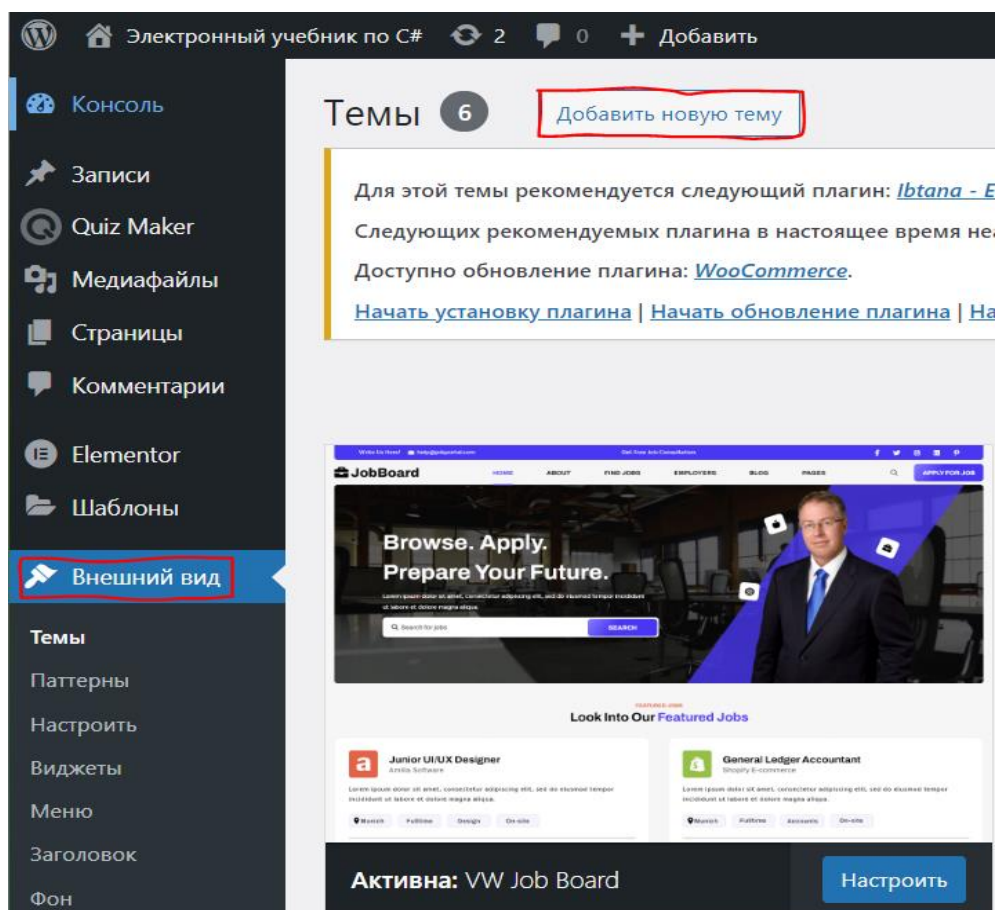


Рисунок 3.2 – Добавление новой темы

Далее, нужно ввести в поиск название темы и установить её.
Поиск темы изображено на рисунке 3.3

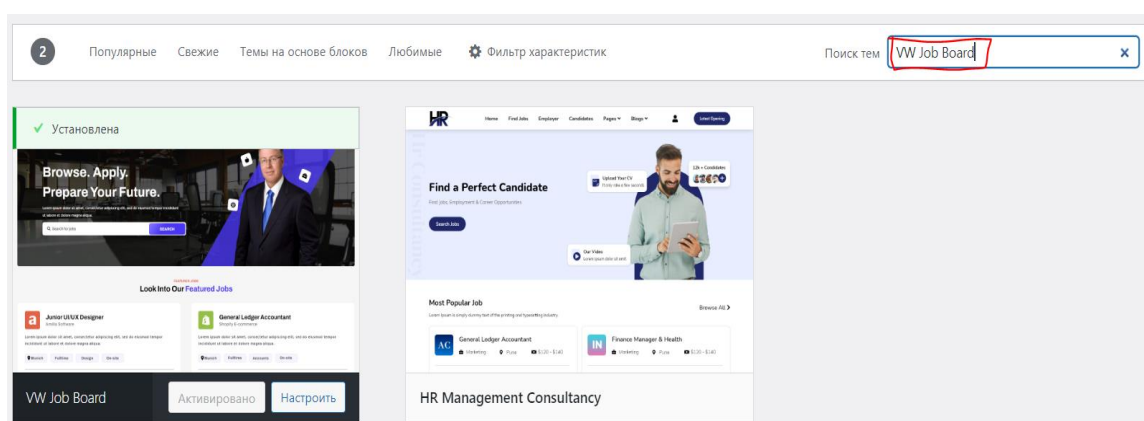


Рисунок 3.3 – Поиск темы

3.1.3 Настройка шапки сайта

В настройках самой темы можно настроить шапку страницы, подвал, шрифты и многое другое.

Для настройки шапки страницы нужно зайти во вкладку «Свойства сайта». Настройка шапки сайта изображена на рисунке 3.4.

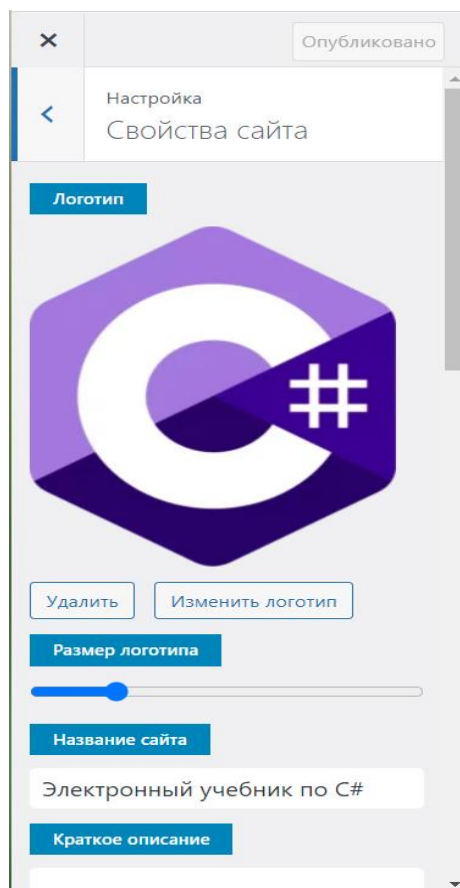


Рисунок 3.4 – Настройка шапки сайта

3.1.4 Настройка подвала

Чтобы настроить подвал, нужно перейти во вкладку «Настройки главной страницы» и перейти во вкладку «Настройки подвала».

Настройка подвала изображена на рисунке 3.5.

×

Опубликовать ⚙

<

Настройка ▸ Настройки главной ...
Настройки подвала

For more options of the footer section

GO PRO

Подвал: показать/скрыть

ON ☐

Footer Heading Font Size

25

Heading Letter Spacing

1

Heading Text Transform

☐ Верхний регистр
☒ Заглавные буквы
☐ Нижний регистр

Heading Font Weight

600 ▾

Рисунок 3.5 – Настройка подвала

3.1.5 Настройка шрифтов

Для настройки шрифтов нужно перейти во вкладку «Типографика темы».

					УП ТРПО 2-40 01 01.35.40.02.24 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		22

Настройка шрифтов изображена на рисунке 3.6.

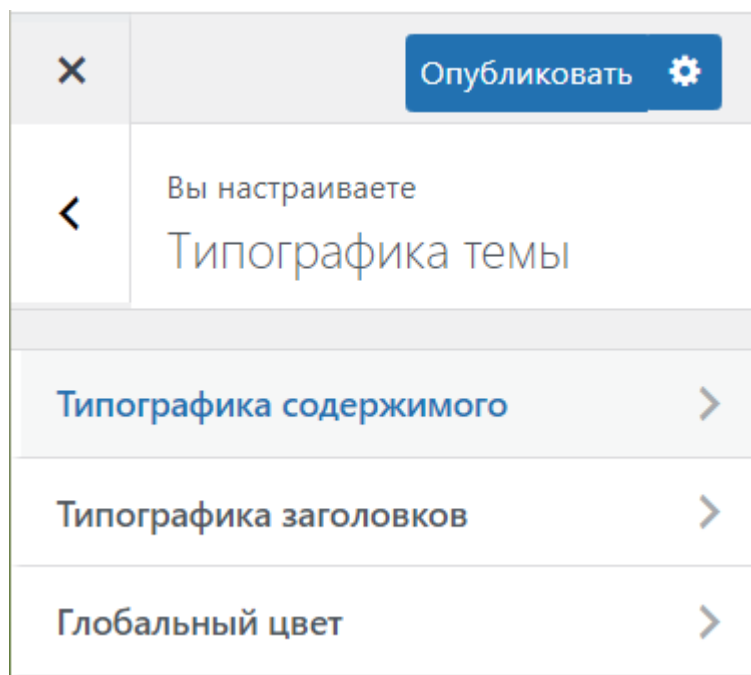


Рисунок 3.6 – Настройка шрифтов

3.1.6 Создание тестов

3.1.6.1 Установка плагина

Для создания тестов был применён плагин Quiz Maker.

Для того, чтобы создать тест в Quiz Maker, нужно сначала установить сам плагин. Для этого, нужно перейти во вкладку «Плагины» и нажать на кнопку «Добавить плагин». Затем, нужно ввести в поиск название плагина Quiz Maker и когда будет найден нужный плагин, нужно нажать на «Установить».

После установки на главной боковой панели WordPress должна располагаться вкладка Quiz Maker. При переходе туда, нужно создать новую викторину.

Quiz Maker изображён на рисунке 3.7.

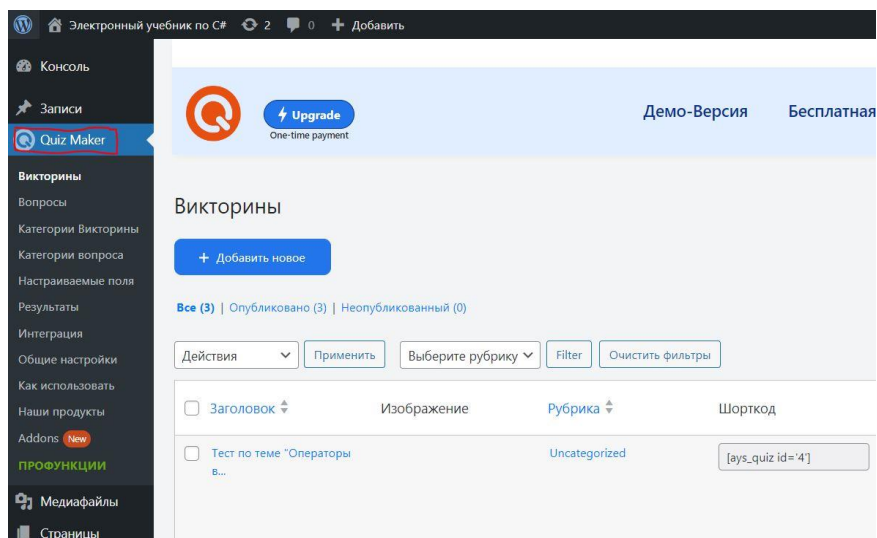


Рисунок 3.7 – Quiz Maker

3.1.6.2 Создание вопросов

Перед тем, как создавать викторину, нужно сначала создать вопросы к самой викторине. Для этого нужно перейти во вкладку «Вопросы» и нажать на кнопку «Добавить новый». Далее, можно создать вопрос с различными вариантами ответов. Также, можно настроить тип выбора ответа(одиночный выбор, флажок, верно/неверно и т.д.).

Вкладка вопросов изображена на рисунке 3.8.

Настройка вопросов изображена на рисунке 3.9.

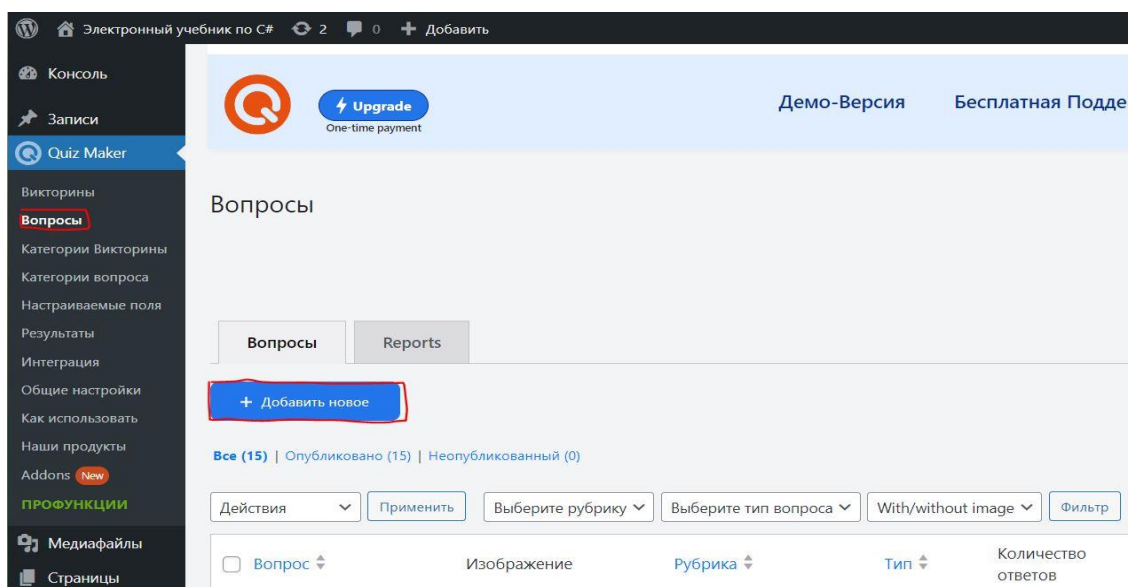


Рисунок 3.8 – Вкладка вопросов

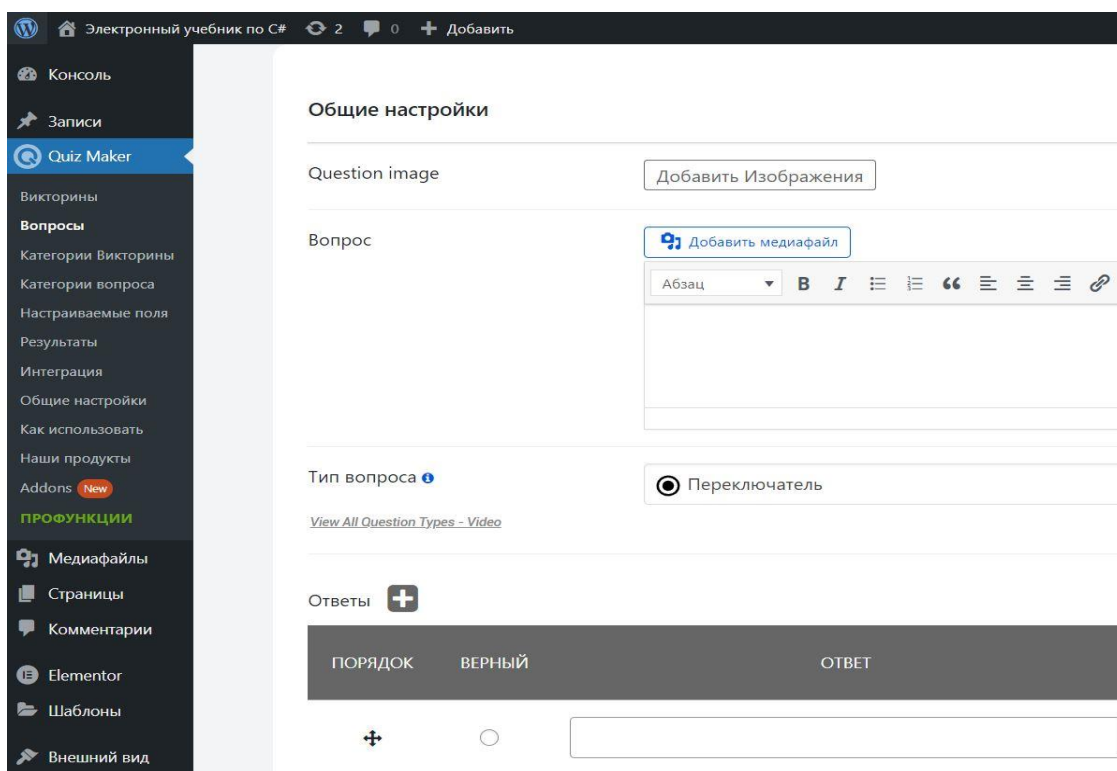


Рисунок 3.9 – Настройка вопросов

3.1.6.3 Создание теста

После того, как вопросы были созданы, можно уже создавать саму викторину, куда можно добавить сами вопросы.

Добавление вопросов в тест изображено на рисунке 3.10.

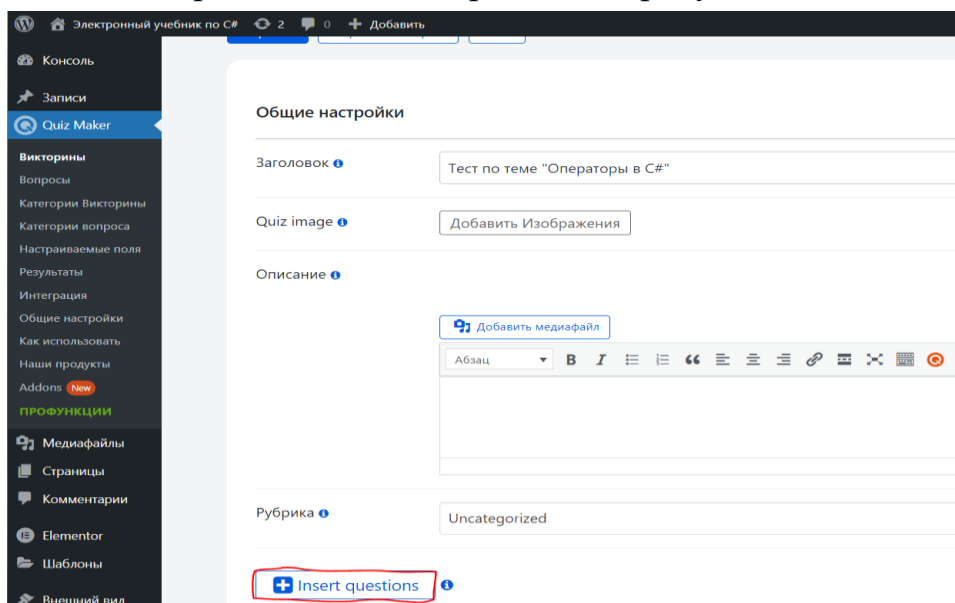


Рисунок 3.10 – Добавление вопросов в тест

3.1.6.4 Настройка внешнего вида теста

Для того, чтобы изменить внешний вид теста, нужно при создании теста перейти во вкладку «Стили». Там можно настроить стиль теста, цвета кнопок, размер окна теста и т.д.

Настройка внешнего вида теста изображена на рисунке 3.11.

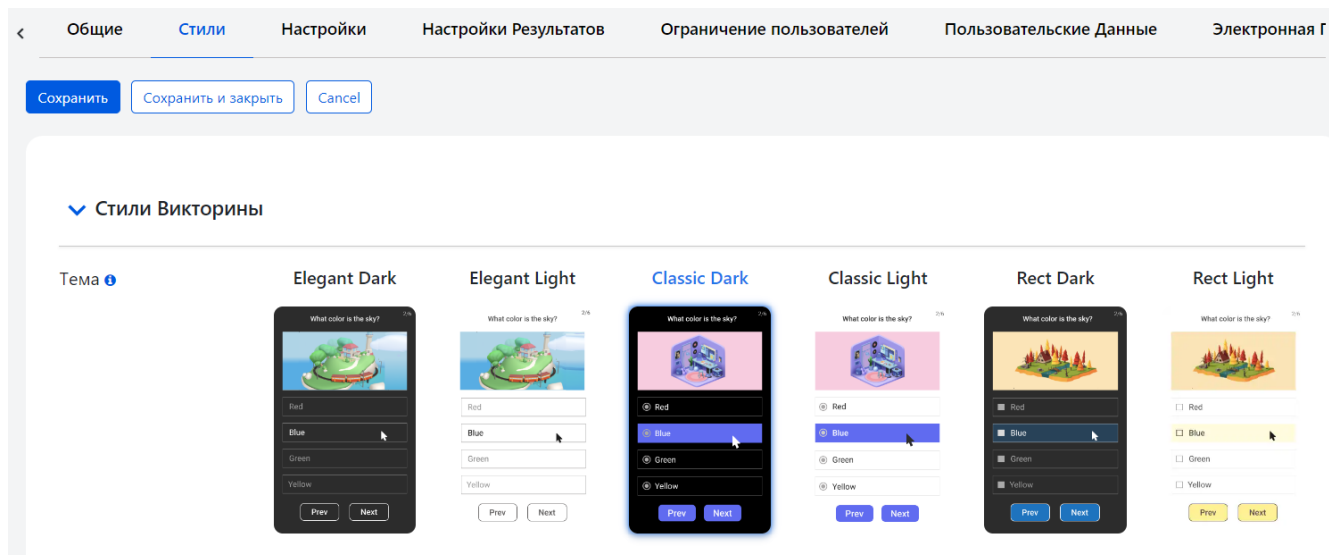


Рисунок 3.11 – Настройка внешнего вида теста

4 Тестирование

4.1 Тесты на использование

При разработке данного программного продукта многие недоработки были исправлены на этапе реализации проекта. Было проведено тщательное функциональное тестирование. Функциональное тестирование должно гарантировать работу всех элементов сайта в автономном режиме.

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

Расписание работ над проектом представлено в таблице 4.1.

Таблица 4.1 – Расписание работ над проектом

Имя	Дата	Деятельность	Продолжительность, ч
Баулин Матвей	22.11.2024	Разработка тестов	3
Баулин Матвей	30.11.2024	Тестирование web-сайта	3
Баулин Матвей	02.12.2024	Составление отчётов о найденных дефектах	2
Баулин Матвей	02.12.2024	Исправление найденных ошибок	1
Баулин Матвей	03.12.2024	Проведение регрессивного тестирования	3
Баулин Матвей	06.12.2024	Составление отчёта о результатах тестирования	2

4.2 Отчёт о результатах тестирования

Статистика по всем дефектам представлена в таблице 4.2.

Таблица 4.2 – Статистика по всем дефектам

Статус	Количество	Важность			
		Низкая	Средняя	Высокая	Критическая
Найдено	2	0	1	1	0
Исправлено	0	0	0	0	0
Проверено					
Открыто заново	0	0	0	0	0
Отклонено	1	0	1	1	0

По результатам тестирования все элементы были проверены и было найдено 2 дефекта, которые невозможно было исправить – кривая адаптация под мобильный телефон и обрезание текста в разделе с теоретическим материалом. Остальные компоненты программного продукта работают как следует.

5 Применение

5.1 Руководство пользователя

Данный программный продукт является электронным средством обучения языку программирования С#.

Данный программный продукт предназначен для обучения такому языку программирования С#. Также, им может пользоваться кто угодно и когда угодно.

Данный программный продукт состоит из главной страницы, которая состоит из шапки сайта, подвала, кнопок-переключателей, гиперссылок и списка теоретического материала, упражнений и тестов.

Главная страница изображена на рисунке 5.1.

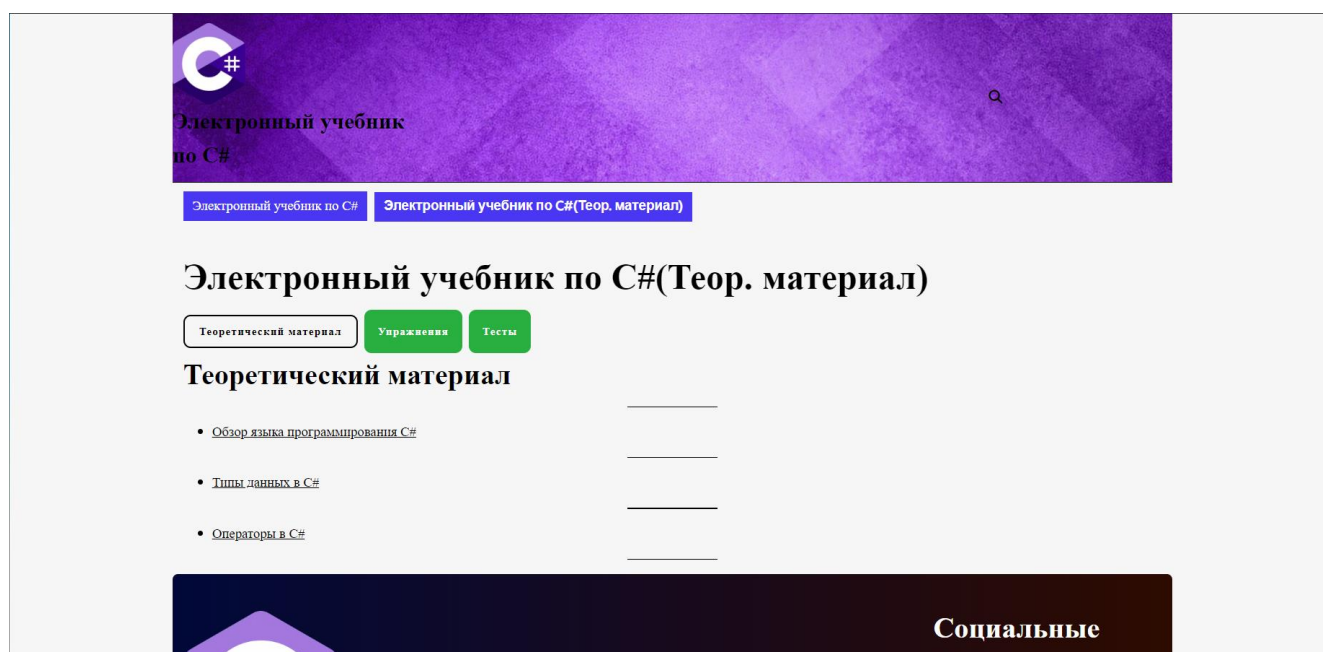


Рисунок 5.1 – Главная страница сайта

Если нажать на кнопку «Упражнения» или на кнопку «Далее», то будет осуществлён переход на страницу со списком упражнений или со списком тестов.

Страница со списком упражнений изображён на рисунке 5.2.

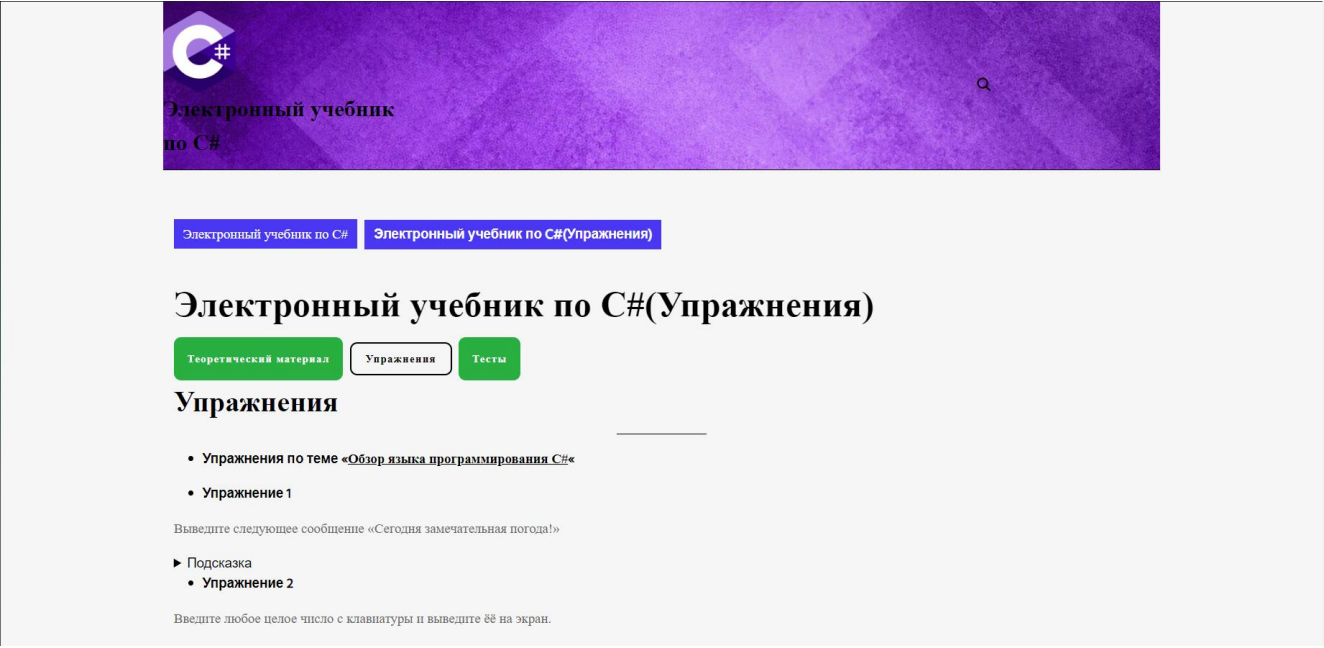


Рисунок 5.2 – Страница со списком упражнений

Если нажать на гиперссылку названия темы, будет осуществлён переход на страницу с теоретическим материалом, упражнениями и тестом по соответствующей теме.

Страница «Обзор языка программирования C#» изображена на рисунке 5.3.

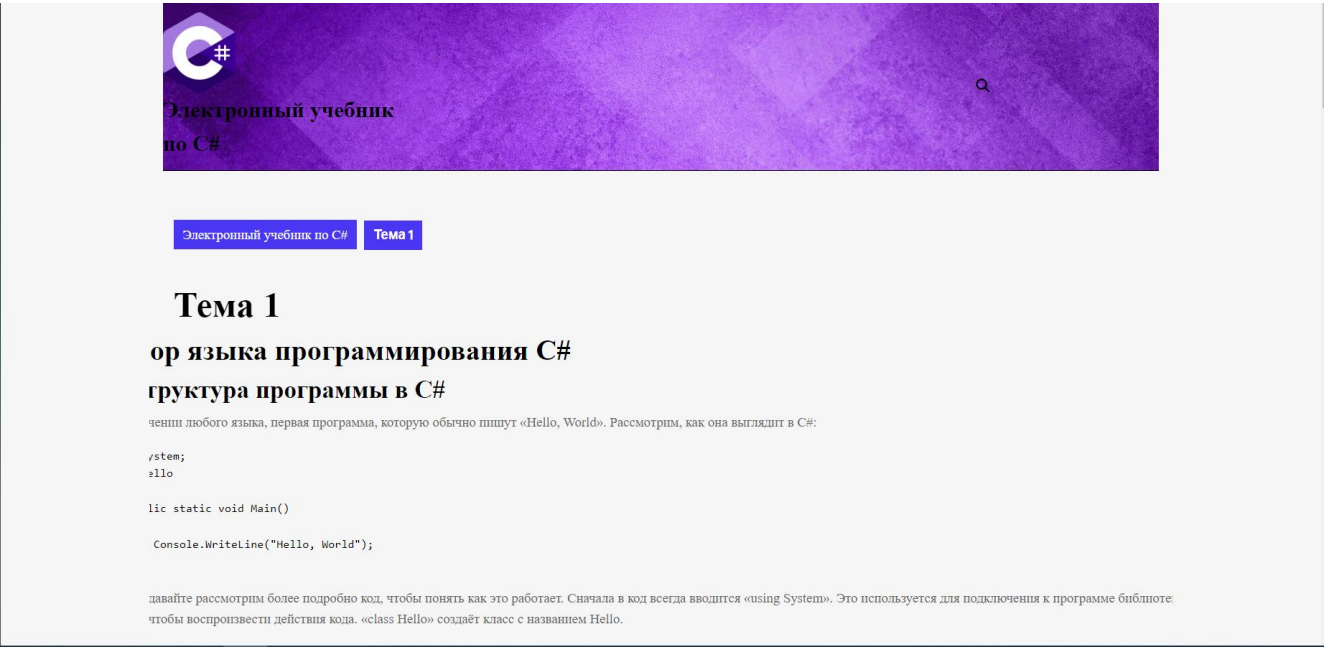


Рисунок 5.3 – Страница «Обзор языка программирования C#»

На этой же странице, помимо просмотра теоретического материала, можно просматривать упражнения для самоконтроля и проходить тест по пройденной теме.

Раздел упражнений выглядит в виде списка упражнений, каждый пункт которого состоит из условия упражнения и подсказки, которая изображена в виде выпадающего списка, чтобы подсказка не была сразу видна пользователю.

Раздел теста содержит в себе окно теста, в котором можно пройти тест для самоконтроля по пройденной теме.

Окно теста изображено на рисунке 5.4.

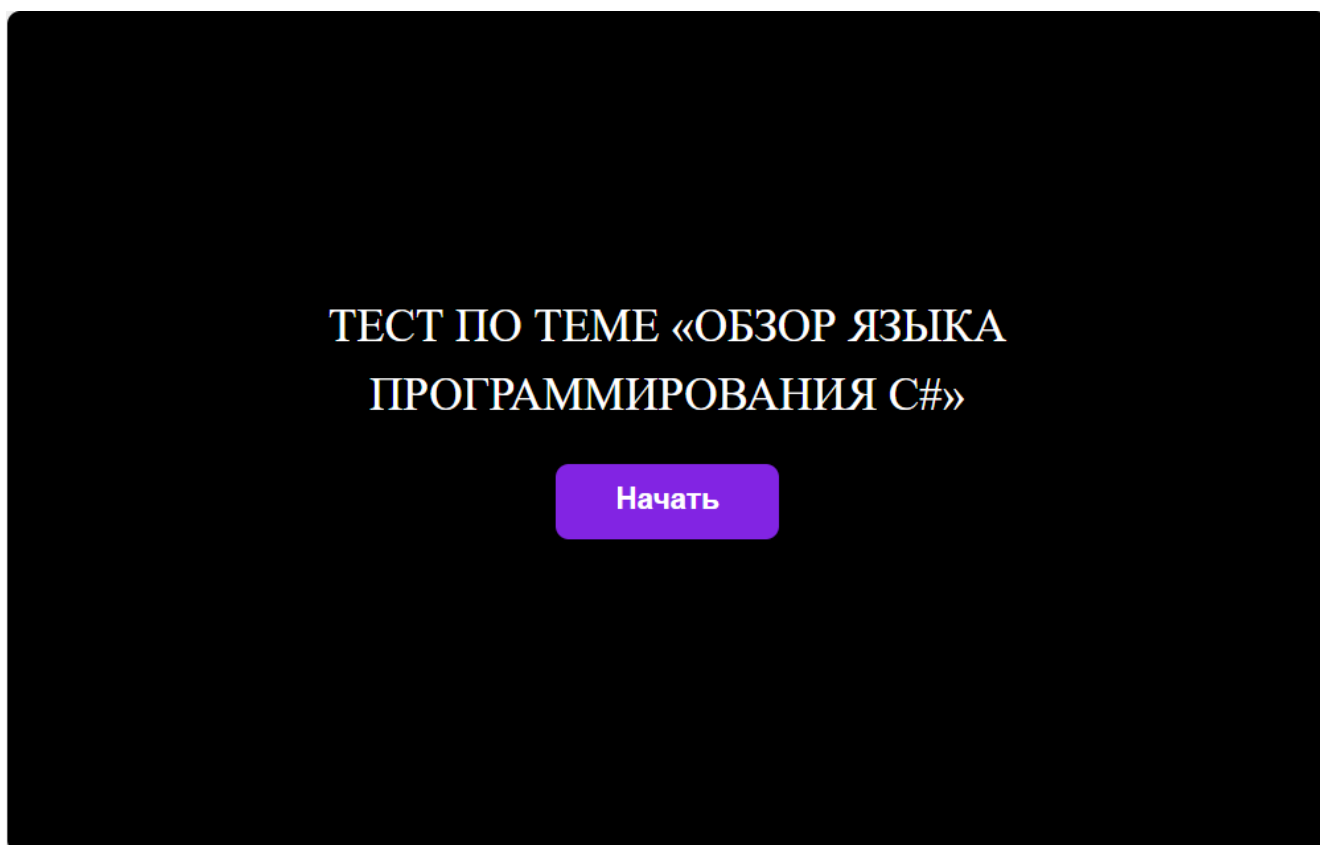


Рисунок 5.4 – Окно теста

При прохождении теста выводится результат теста с оценкой в виде процента и со списком ответов на каждый вопрос.

Результат теста изображён на рисунке 5.5.

					УП ТРПО 2-40 01 01.35.40.02.24 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		30

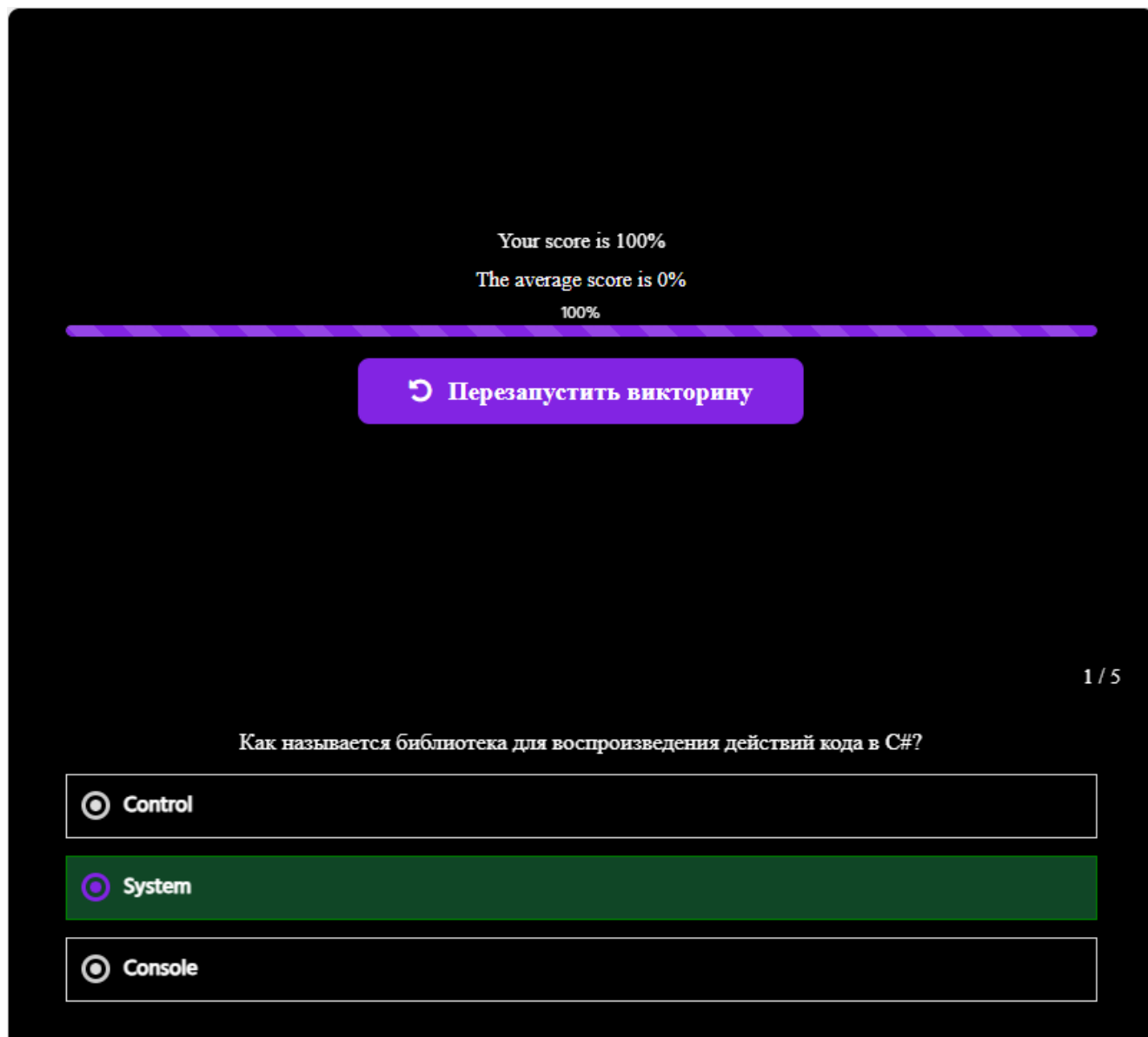


Рисунок 5.5 – Результат теста

После раздела теста для самоконтроля находится кнопка «На главную страницу», которая возвращает на главную страницу и подвал сайта и при некотором расстоянии от шапки сайта создаётся кнопка со стрелочкой «↑», при нажатии на которую осуществляется переход на верх сайта.

Подвал сайта и кнопка «↑» изображены на рисунке 5.6.

					УП ТРПО 2-40 01 01.35.40.02.24 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		31

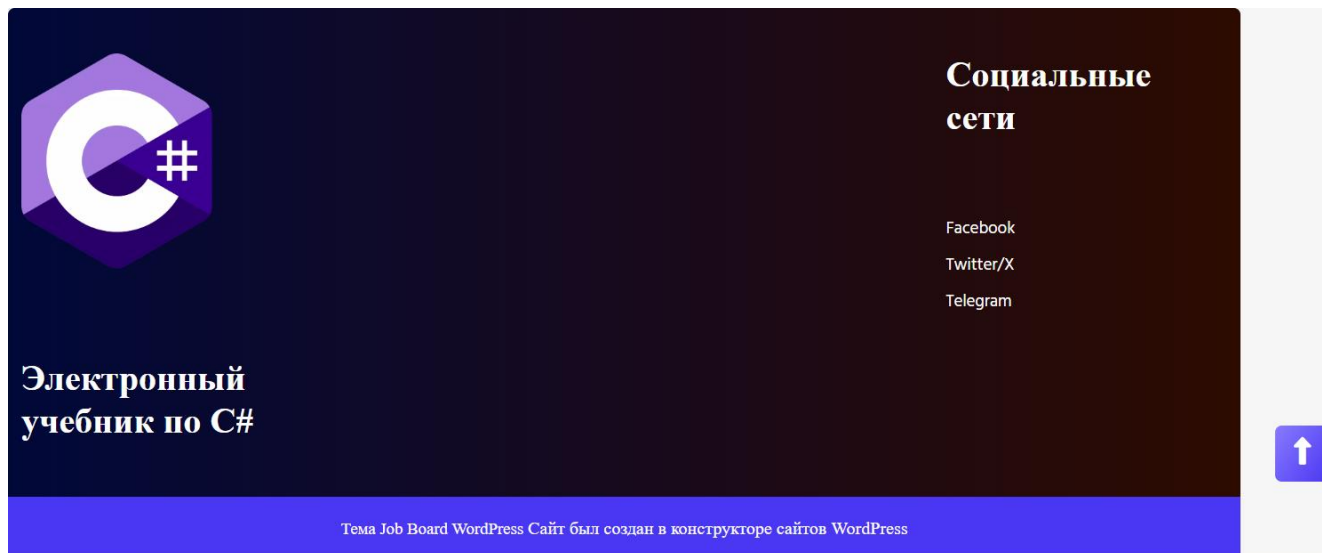


Рисунок 5.6 - Подвал сайта и кнопка «↑»

Также, в шапке сайта есть раздел поиска, где можно будет осуществить поиск определённой темы. В результате будут выведены разделы, в которых содержится найденная тема.

Результаты поиска «Типы данных в C#» изображены на рисунке 5.7.

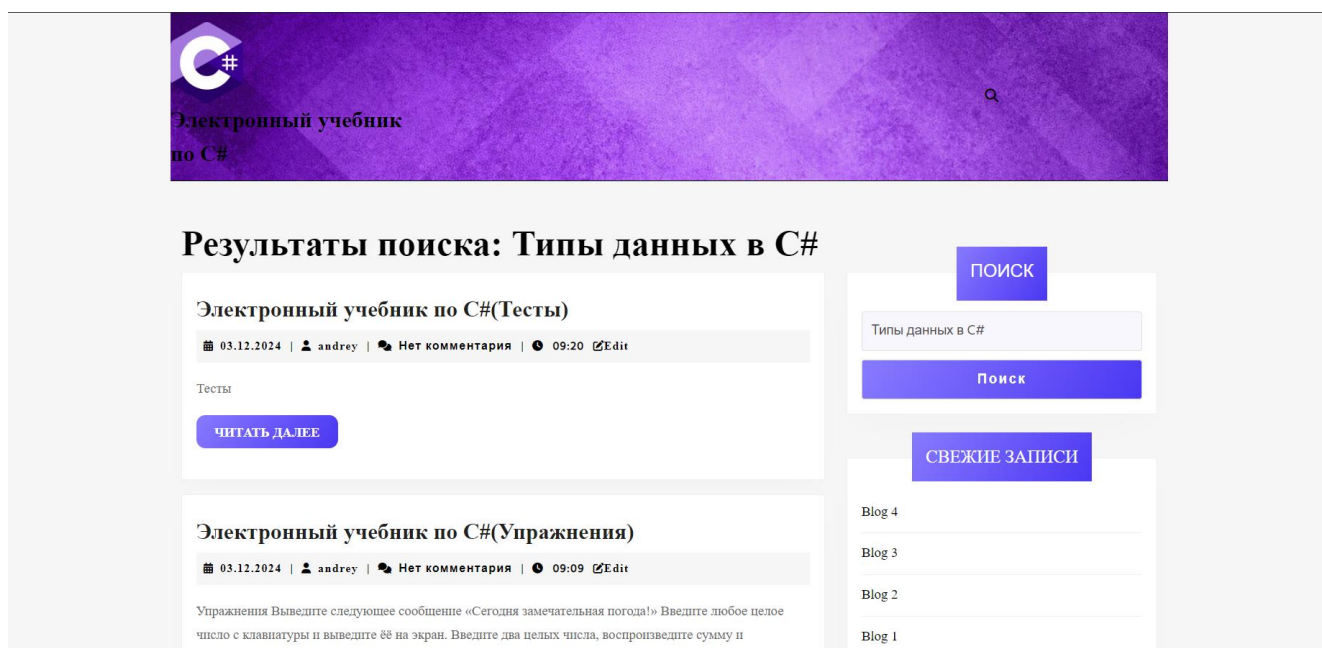


Рисунок 5.7 – Результаты поиска «Типы данных в C#»

					УП ТРПО 2-40 01 01.35.40.02.24 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		32

Заключение

В итоге проведённой работы по разработке электронного средства обучения по языку программирования С# был создан сайт, содержащий в себе 3 классические темы по С#, упражнения и тесты для самоконтроля.

Компоненты сайта работают отлично, за исключением двух дефектов, которые были указаны в разделе 4.2.

Этим сайтом можно пользоваться в любой момент, т.к. найдётся желающий узнать чего-то нового про такой язык программирования, как С#, который является одним из самых популярных языков программирования на сегодняшний день.

					УП ТРПО 2-40 01 01.35.40.02.24 ПЗ	Лист
Изм.	Лист	№докум.	Подпись	Дата		33

Список использованных источников

1. Программирование на языке С# / И.Л. Александрова, Д.Н. Тумаков [Электронный ресурс]. – Режим доступа: https://kpfu.ru/staff_files/F_2033489788/Tumakov_Programmirovanie_na_yazyke_C_.pdf - Дата доступа: 01.11.2024.

					УП ТРПО 2-40 01 01.35.40.02.24 ПЗ	Лист
						34
Изм.	Лист	№докум.	Подпись	Дата		