

**Минобрнауки России**  
**Юго-Западный государственный университет**

Кафедра программной инженерии

**ОТЧЕТ**

о преддипломной практике

наименование вида и типа практики

на (в) Юго-Западном государственном университете

наименование предприятия, организации, учреждения

Студента 4 курса, группы ПО-916

курса, группы

Иванова Ивана Ивановича

фамилия, имя, отчество

Руководитель практики от  
предприятия, организации,  
учреждения

Оценка \_\_\_\_\_

должность, звание, степень

фамилия и. о.

подпись, дата

Руководитель практики от  
университета

Оценка \_\_\_\_\_

должность, звание, степень

фамилия и. о.

подпись, дата

Члены комиссии

подпись, дата

фамилия и. о.

подпись, дата

фамилия и. о.

Курск, 2023 г.

## СОДЕРЖАНИЕ

1	Анализ предметной области	3
1.1	Характеристика предприятия и его деятельности	3
1.2	Аддитивные технологии, их классификация	4
2	Техническое задание	5
2.1	Основание для разработки	5
2.2	Цель и назначение разработки	5
2.3	Требования пользователя к интерфейсу web-сайта	5
2.4	Моделирование вариантов использования	6
2.5	Требования к оформлению документации	7
3	Технический проект	8
3.1	Общая характеристика организации решения задачи	8
3.2	Обоснование выбора технологии проектирования	8
3.2.1	Описание используемых технологий и языков программирования	8
3.2.2	Язык программирования PHP	9
3.2.3	Язык программирования JavaScript	9
3.3	Диаграмма компонентов и схема обмена данными между файлами компонента	10
3.4	Диаграмма размещения	12
3.5	Содержание информационных блоков. Основные сущности	13
4	Рабочий проект	15
4.1	Классы, используемые при разработке сайта	15
4.2	Тестирование разработанного web-сайта	16
	СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	19

# **1 Анализ предметной области**

## **1.1 Характеристика предприятия и его деятельности**

Технология трёхмерной печати появилась в конце 80-х гг. XX в. Пионером в этой области является компания 3D Systems, которая разработала первую коммерческую стереолитографическую машину – SLA – Stereolithography Apparatus (1986 г.). До середины 90-х гг. она использовалась в научно-исследовательской и опытноконструкторской деятельности, связанной с оборонной промышленностью. Первые лазерные машины – сначала стереолитографические (SLA-машины), затем порошковые (SLS-машины) – были чрезмерно дороги, а выбор модельных материалов скромный. Широкое распространение цифровых технологий в области проектирования (CAD), моделирования и расчётов (CAE) и механообработки (CAM) стимулировало развитие технологий 3D-печати.

Термин “аддитивные технологии” (АТ) означает изготовление изделия путем добавления. АТ являются новыми методами в производстве различного рода изделий. Применение данных технологий допускает как создание изделий с нуля, так и обработку уже имеющихся. Сегодня трудно найти отрасль производства, где бы ни применялись 3D-принтеры: с их помощью изготавливаются детали самолетов, космических аппаратов, подводных лодок, инструменты, протезы и др.

АТ предполагают изготовление (построение) физического объекта (детали) методом послойного нанесения материала, в отличие от традиционных методов формирования детали, за счёт удаления материала из массива заготовки.

АТ охватывают все новые сферы деятельности человека. Дизайнеры, архитекторы, кондитеры, археологи, астрономы, палеонтологи и представители других профессий используют 3D-принтеры для реализации различных идей и проектов.

Деятельность отраслевого интегратора “Русатом – Аддитивные технологии” охватывает все составляющие аддитивного рынка: производство 3D-принтеров, выпуск оборудования для создания порошков, разработка программного обеспечения и организация центров аддитивного производства. Сегодня аддитивные технологии внедряются в самые сложные и наукоемкие отрасли: атомную промышленность, аэрокосмическую индустрию, медицину, автомобилестроение и многие другие. Применение АТ решает задачи по снижению стоимости, сокращения срока изготовления изделий и обеспечение высокой персонализации деталей.

## **1.2 Аддитивные технологии, их классификация**

Основное преимущество АТ состоит в том, что прототип создается за один прием, а исходными данными для него служит геометрическая модель детали. В итоге отпадает необходимость в планировании последовательности технологических процессов, специальному оборудованию для обработки материалов, транспортировке от станка к станку и т. д.

Экструзионная печать. Включает такие методы, как послойное наплавление и многоструйная печать.

Стереолитография. Стереолитографические принтеры используют специальные жидкие материалы, называемые “фотополимерными смолами”.

Ламинирование. Слои материала наклеиваются друг на друга и обрезаются по контурам цифровой модели с помощью лазера или лезвия.

## **2 Техническое задание**

### **2.1 Основание для разработки**

Основанием для разработки является задание на выпускную квалификационную работу бакалавра “Разработка web-сайта “Русатом –Аддитивные технологии” на платформе 1С-Битрикс”.

### **2.2 Цель и назначение разработки**

Основной задачей выпускной квалификационной работы является разработка и внедрение web-сайта для продвижения компании ООО «Русатом – Аддитивные технологии».

Посредством внедрения web-сайта планируется устранить существующие недостатки в компании. Исходя из этого, основную цель предлагается рассмотреть в разрезе двух групп подцелей.

Задачами данной разработки являются:

- создание информационных разделов сайта;
- реализация формы для обратной связи;
- реализация калькулятора расчета стоимости изготовления деталей;
- реализация формы заявки на изготовление деталей;
- создание удобного поиска по сайту.

### **2.3 Требования пользователя к интерфейсу web-сайта**

Сайт должен включать в себя:

- навигацию по разделам;
- авторизацию;
- доступы для администратора, редактора, исполнителя по заявкам с форм.

Композиция шаблона сайта представлена на рисунке 2.1.

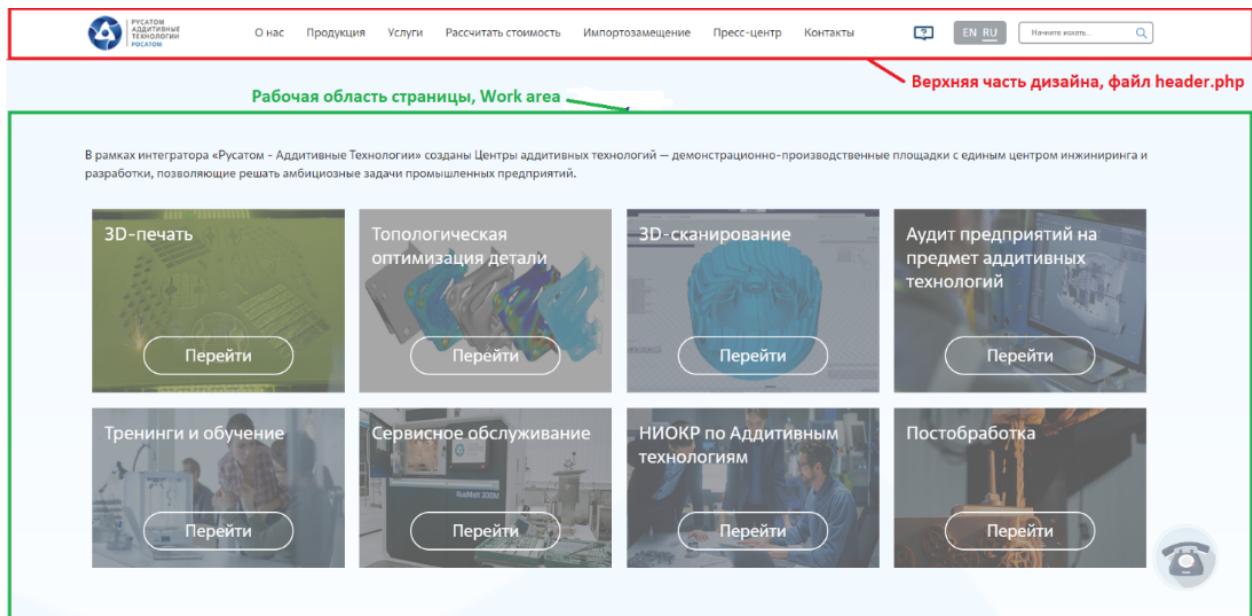


Рисунок 2.1 – Композиция шаблона сайта

## 2.4 Моделирование вариантов использования

Для разрабатываемого сайта была реализована модель, которая обеспечивает наглядное представление вариантов использования сайта.

Она помогает в физической разработке и детальном анализе взаимосвязей объектов. При построении диаграммы вариантов использования применяется унифицированный язык визуального моделирования UML.

Диаграмма вариантов описывает функциональное назначение разрабатываемой системы. То есть это то, что система будет непосредственно делать в процессе своего функционирования. Она является исходным концептуальным представлением системы в процессе ее проектирования и разработки. Проектируемая система представляется в виде ряда прецедентов, предоставляемых системой актерам или сущностям, которые взаимодействуют с системой. Актером или действующим лицом является сущность, взаимодействующая с системой извне (например, человек, техническое устройство). Прецедент служит для описания набора действий, которые система предоставляет актеру.

На основании анализа предметной области в программе должны быть реализованы следующие прецеденты:

1. Просмотр информации о компании.
2. Просмотр информации о продукции компании.
3. Просмотр информации об услугах компании, много услуг, длинный список, не поместится никак в одну строку.
4. Поиск по сайту.

## **2.5 Требования к оформлению документации**

Разработка программной документации и программного изделия должна производиться согласно ГОСТ 19.102-77 и ГОСТ 34.601-90. Единая система программной документации.

### **3 Технический проект**

#### **3.1 Общая характеристика организации решения задачи**

Необходимо спроектировать и разработать сайт, который должен способствовать продвижению компании на рынке.

Интернет-сайт представляет собой набор взаимосвязанных электронных страниц, которые сгруппированы по разделам, содержащие текстовую, графическую, а также мультимедийную информацию (изображения, видеоролики и пр.). Сайт располагается в Интернете по определенному адресу – доменному имени сайта в виде [www.имя\\_сайта.ru](http://www.имя_сайта.ru). Каждая страница web-сайта – это текстовый документ, написанный на языке программирования (HTML, CSS, JavaScript и т.д.).

#### **3.2 Обоснование выбора технологии проектирования**

На сегодняшний день информационный рынок, поставляющий программные решения в выбранной сфере, предлагает множество продуктов, позволяющих достигнуть поставленной цели – разработки web-сайта.

##### **3.2.1 Описание используемых технологий и языков программирования**

В процессе разработки web-сайта используются программные средства и языки программирования. Каждое программное средство и каждый язык программирования применяется для круга задач, при решении которых они необходимы.

### **3.2.2 Язык программирования PHP**

PHP – язык для написания сценариев, исполняемых на компьютере web-приложения посредством интерпретации исходного кода [2]. Основное предназначение данного языка – это выполнение на сервере сценариев, создающих динамические web-страницы.

### **3.2.3 Язык программирования JavaScript**

JavaScript – объектно-ориентированный язык программирования для написания сценариев [1]. Чаще всего JavaScript используется для написания сценариев работы с web-страницами, отображаемыми web-браузером. Web-браузер интерпретирует код сценария языка JavaScript, и на основе описанных в сценарии действий производит манипуляции с разметкой web-страницы. Посредством языка JavaScript реализуется возможность программирования на стороне клиента. Предоставляет возможность доступа к элементам разметки web-страницы посредством объектов. При создании сценариев на языке JavaScript приходится сталкиваться с трудностями, связанными с тем, что различные web-браузеры могут по-разному интерпретировать эти сценарии. Серьезные трудности возникают, если какой-либо из браузеров не поддерживает тот или иной объект, метод или свойство. Наиболее практическим способом решения данной проблемы является использование библиотеки jQuery. Данная библиотека реализована на языке JavaScript и расширяет возможности данного языка, нивелируя различия между браузерами.

### 3.3 Диаграмма компонентов и схема обмена данными между файлами компонента

Диаграмма компонентов описывает особенности физического представления разрабатываемой системы. Она позволяет определить архитектуру системы, установив зависимости между программными компонентами, в роли которых может выступать как исходный, так и исполняемый код. Основными графическими элементами диаграммы компонентов являются компоненты, интерфейсы, а также зависимости между ними. На рисунке 3.2 изображена диаграмма компонентов для проектируемой системы. Она включает в себя сервер с операционной системой, на которой установлена система управления содержимым, включающая в себя базу данных и интерфейс. Помимо этого на диаграмме изображен клиентский компьютер с операционной системой, на которой установлен браузер.

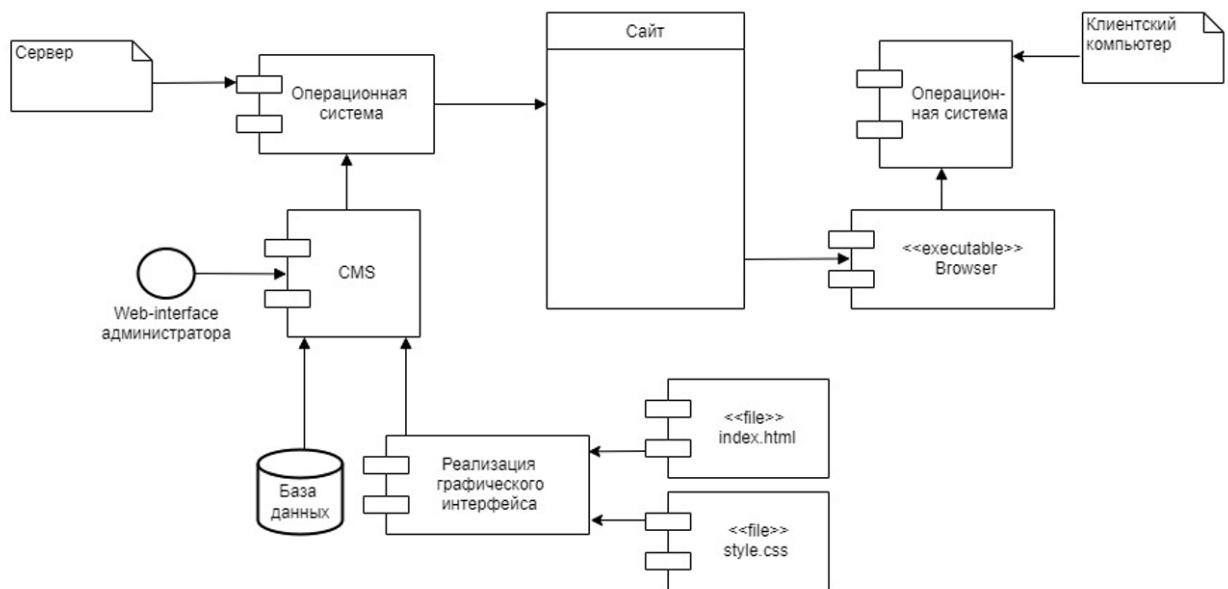


Рисунок 3.2 – Диаграмма компонентов

Любой компонент должен быть вызван в сценарии страницы web-сайта. Web-страница передает данные компоненту в момент вызова последнего.

На рисунке 3.3 представлена схема обмена данными между сценариями компонента при вызове компонента на странице сайта.

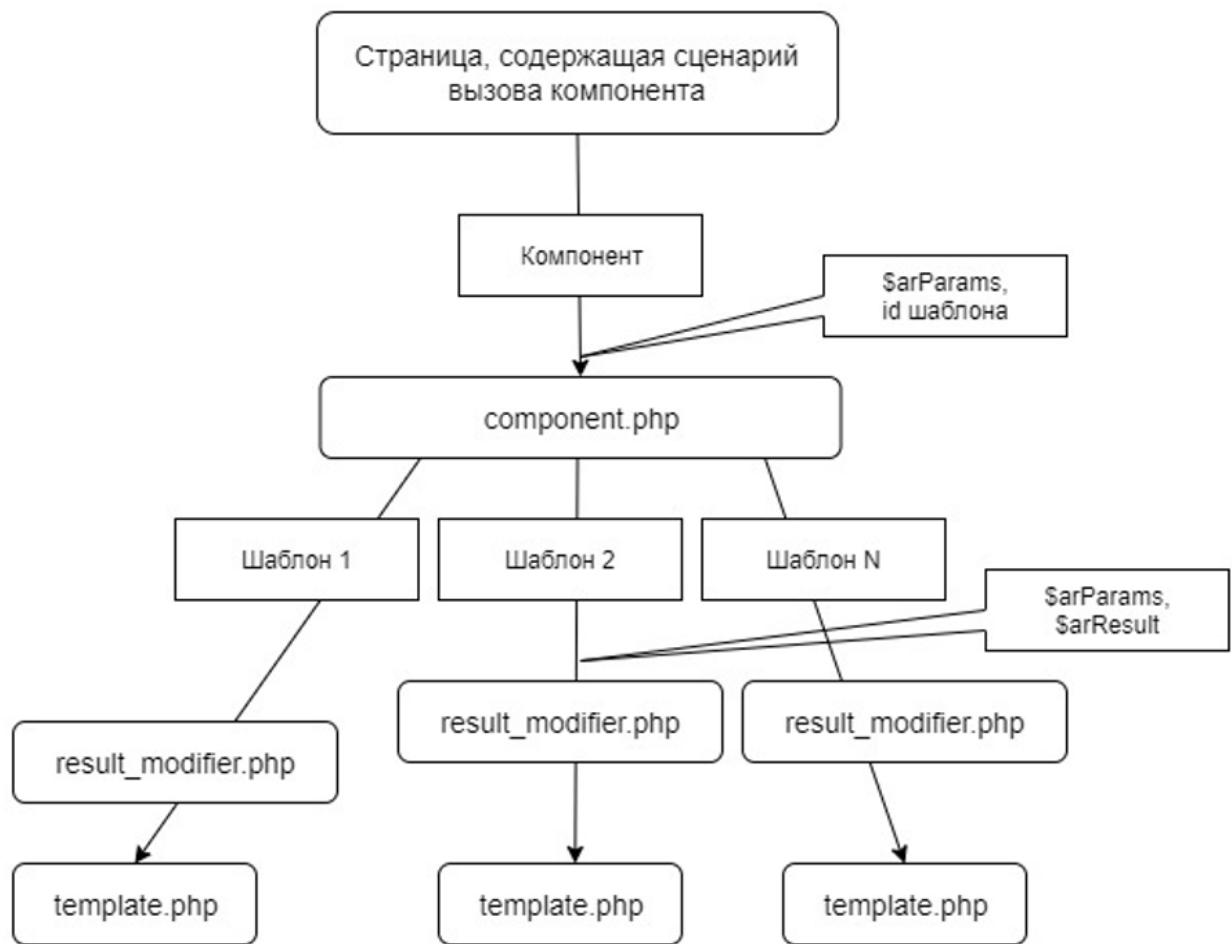


Рисунок 3.3 – Диаграмма компонентов

При вызове компонента в сценарии web-страницы указываются значения параметров компонента, которые далее посредством массива `$arParams` передаются в сценарий файла `component.php`.

В сценарии файла `component.php` посредством метода `IncludeComponentTemplate` класса `CBitrixComponent` происходит вызов одного из шаблонов компонента. Id шаблона также определяется в сценарии страницы web-приложения и неявно для разработчика передается указанный выше метод. Подключается сценарий файла `template.php` одного из шаблонов, в который передается, возможно, измененный в сценарии `component.php` массив `$arParams` и, также, сформированный в сценарии `component.php` массив `$arResult`. Оба этих массива доступны также и в файле `result_modifier.php`, который подключается перед подключением файла `template.php`.

Работа компонента заканчивается в момент завершения работы сценария файла component.php, т.е. возможно выполнить действия уже после подключения шаблона. Однако, если массив \$arResult будет изменен в сценарии шаблона, в сценарий файла компонента component.php измененные данные переданы не будут.

### 3.4 Диаграмма размещения

Диаграмма размещения (рис. 3.4) отражает физические взаимосвязи между программными и аппаратными компонентами системы. Она является хорошим средством для показа маршрутов перемещения объектов и компонентов в распределенной системе.

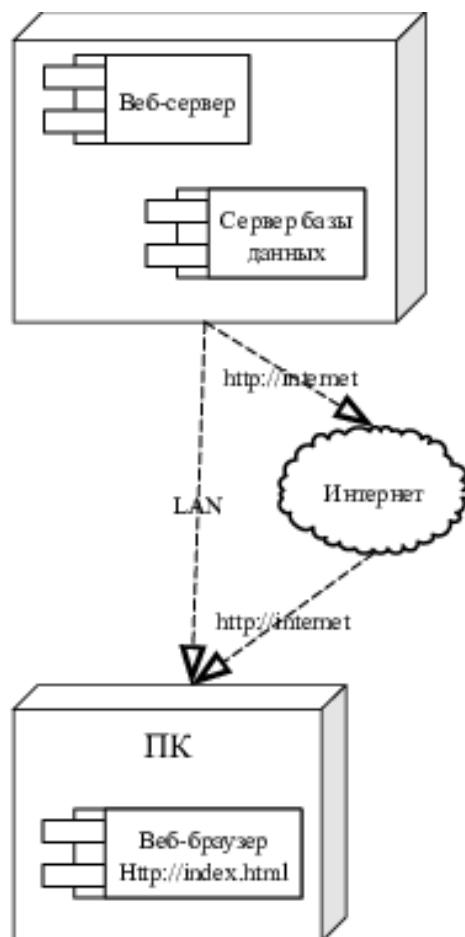


Рисунок 3.4 – Диаграмма размещения

### 3.5 Содержание информационных блоков. Основные сущности

Проанализировав требования, можно выделить шесть основных сущностей:

- “Новости”;
- “Продукция”;
- “Услуги”.

В состав сущности “Новости” можно включить атрибуты, представленные в таблице 3.1.

Таблица 3.1 – Атрибуты сущности “Новости”

Поле	Тип	Обязательное
1	2	3
ID	string	true
Активность	char	true

В состав сущности “Продукция” и “Услуги” можно включить атрибуты, представленные в таблице 3.2.

Таблица 3.2 – Атрибуты сущности “Продукция” и “Услуги”

Поле	Тип	Обязательное
ID	string	true
Активность	char	true
Активность	char	true
Активность	char	true

В системе предусмотрен внутренний механизм связи между разделами и элементами информационных блоков, поэтому введения дополнительных идентификаторов при реализации связей между сущностями не предполагается.

Экземпляры сущностей реализуются в информационных блоках посредством элементов, атрибуты сущности – посредством полей и свойств элемента.

## 4 Рабочий проект

### 4.1 Классы, используемые при разработке сайта

Можно выделить следующий список классов и их методов, использованных при разработке web-приложения (таблица 4.3).

Таблица 4.3 – Описание классов Bitrix, используемых в приложении

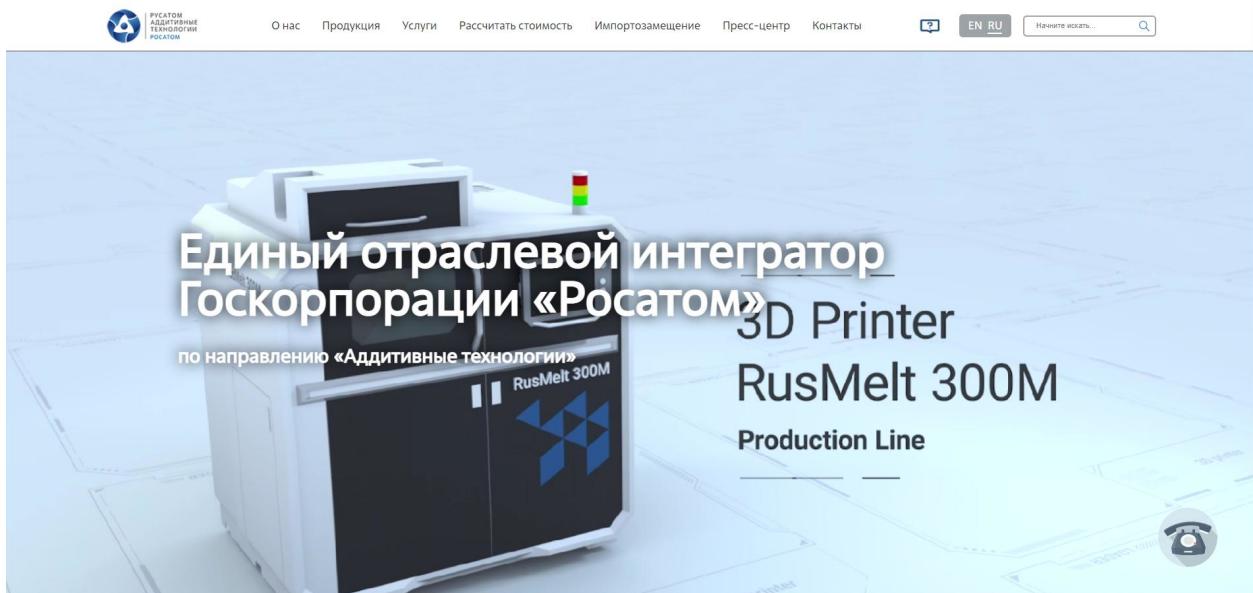
Название класса	Модуль, к которому относится класс	Описание класса	Методы
1	2	3	4
CMain	Главный модуль	CMain – главный класс страницы web-приложения. После одного из этапов по загрузке страницы в сценарии становится доступным инициализированный системой объект данного класса с именем \$APPLICATION	void ShowTitle(string property_code = «title», bool strip_tags = true) Выводит заголовок страницы void SetTitle(string title) Устанавливает заголовок страницы void ShowCSS(bool external = true, bool XhtmlStyle = true) Выводит таблицу стилей CSS страницы

### Продолжение таблицы 4.3

1	2	3	4
CFile	Главный модуль	CFile – Класс для работы с файлами и изображениями	array GetFileArray (int file_id) Метод возвращает массив, содержащий описание файла (путь к файлу, имя файла, размер) с идентификатором file_id

## 4.2 Тестирование разработанного web-сайта

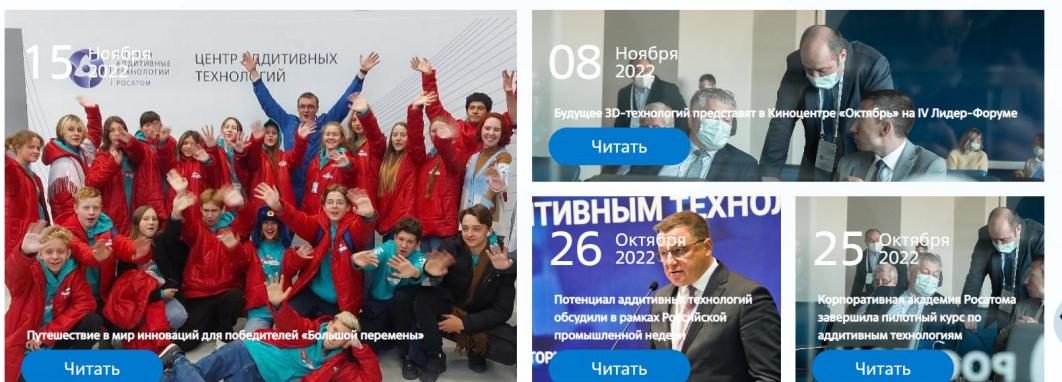
На рисунке 4.5 представлена главная страница сайта «Русатом – Аддитивные технологии».



ООО «РусАТ» обладает всеми необходимыми ресурсами, чтобы помочь нашим заказчикам оценить, протестировать и осуществить масштабное внедрение аддитивного производства на предприятие.

## Новости

Новости



## Области применения

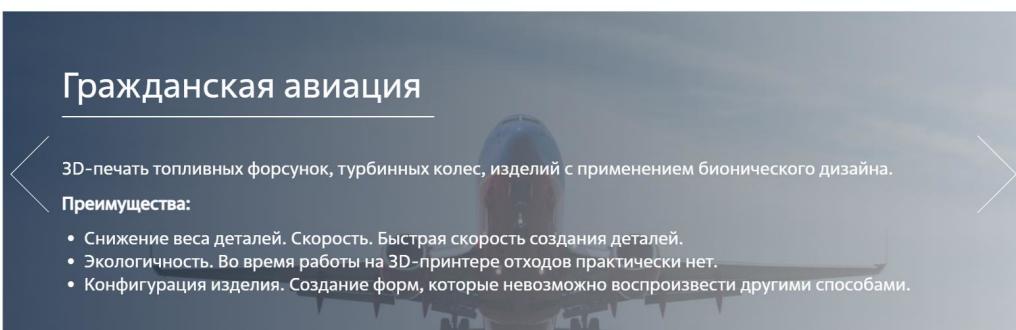


Рисунок 4.5 – Главная страница сайта «Русатом – Аддитивные технологии»

На рисунке 4.6 представлен динамический вывод заголовков, включающий в себя искомые фразы при поиске фраз.

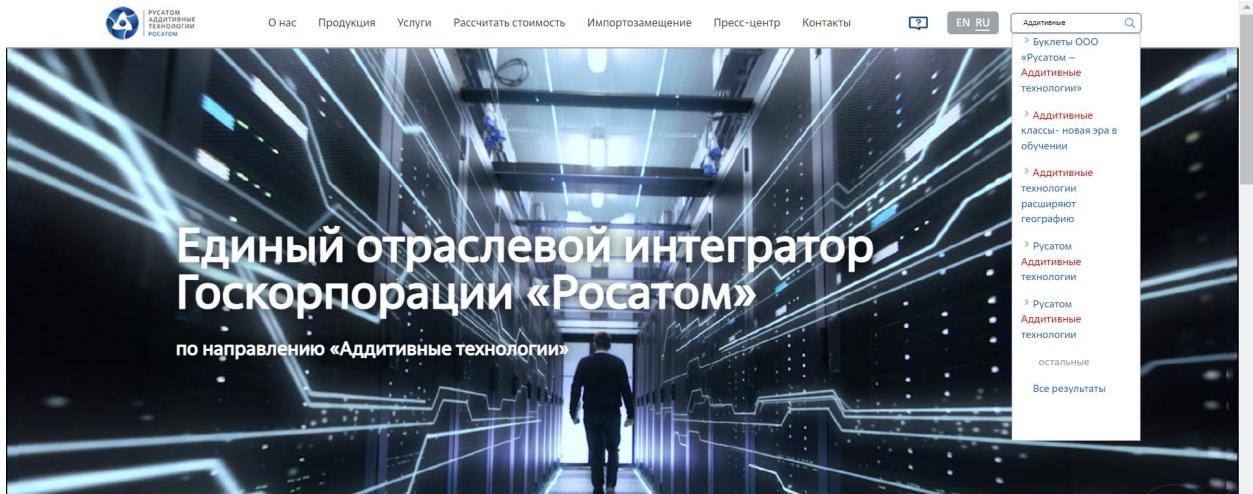


Рисунок 4.6 – Динамический вывод заголовков

На рисунке 4.7 представлен ввод данных для публикации новости.

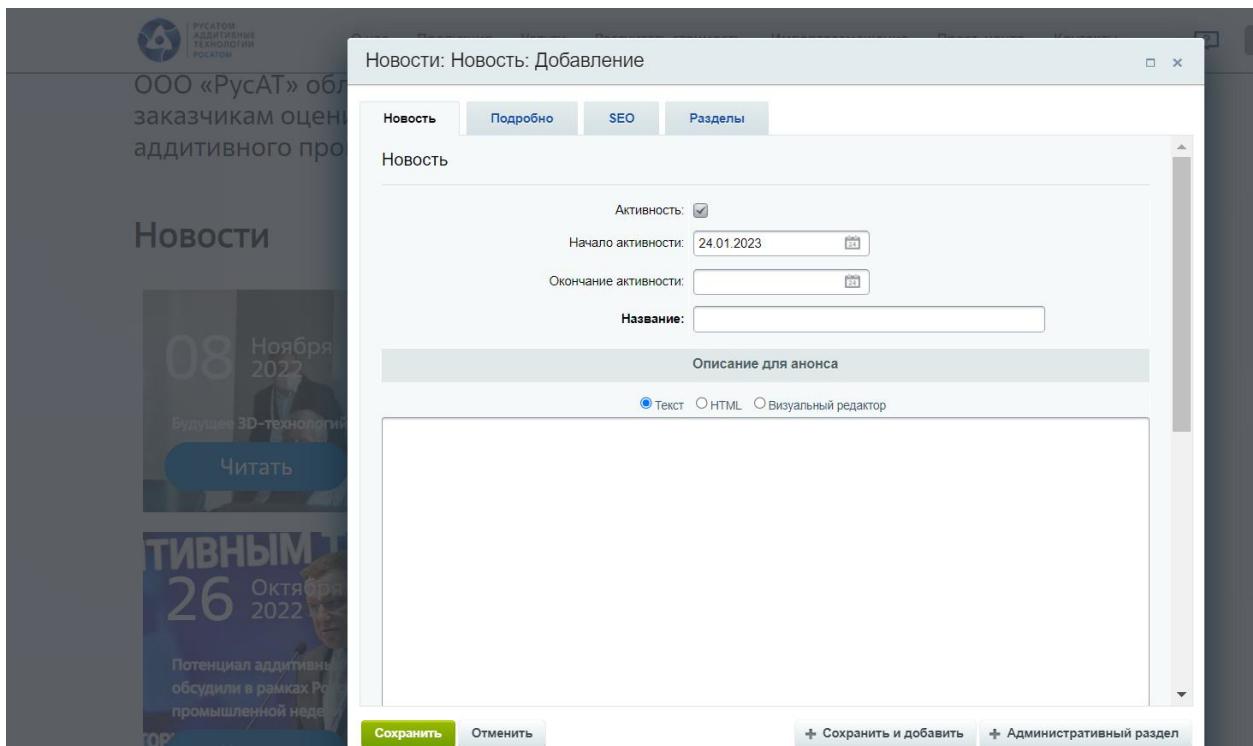


Рисунок 4.7 – Ввод данных для публикации очень-очень длинной, интересной и полезной новости

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Фримен, А. Практикум по программированию на JavaScript / А. Фримен.– Москва : Вильямс, 2013. – 960 с. – ISBN 978-5-8459-1799-7. – Текст : непосредственный.
2. Бретт, М. PHP и MySQL. Исчерпывающее руководство / М. Бретт. – Санкт-Петербург : Питер, 2016. – 544 с. – ISBN 978-5-496-01049-8. – Текст : непосредственный.
3. Веру, Л. Секреты CSS. Идеальные решения ежедневных задач / Л. Веру. – Санкт-Петербург : Питер, 2016. – 336 с. – ISBN 978-5-496-02082-4. – Текст : непосредственный.
4. Гизберт, Д. PHP и MySQL / Д. Гизберт. – Москва : НТ Пресс, 2013. – 320 с. – ISBN 978-5-477-01174-2. – Текст : непосредственный.
5. Голдстайн, А. HTML5 и CSS3 для всех / А. Голдстайн, Л. Лазарис, Э. Уэйл. – Москва : Вильямс, 2012. – 368 с. – ISBN 978-5-699-57580-0. – Текст : непосредственный.
6. Дэкетт, Д. HTML и CSS. Разработка и создание веб-сайтов / Д. Дэкетт. – Москва : Эксмо, 2014. – 480 с. – ISBN 978-5-699-64193-2. – Текст : непосредственный.
7. Макфарланд, Д. Большая книга CSS / Д. Макфарланд. – Санкт-Петербург : Питер, 2012. – 560 с. – ISBN 978-5-496-02080-0. – Текст : непосредственный.
8. Лоусон, Б. Изучаем HTML5. Библиотека специалиста / Б. Лоусон, Р. Шарп. – Санкт-Петербург : Питер, 2013 – 286 с. – ISBN 978-5-459-01156-2. – Текст : непосредственный.
9. Титтел, Э. HTML5 и CSS3 для чайников / Э. Титтел, К. Минник. – Москва : Вильямс, 2016 – 400 с. – ISBN 978-1-118-65720-1. – Текст : непосредственный.
10. Титтел, Э. HTML5 и CSS3 для чайников / Э. Титтел, К. Минник. – Москва : Вильямс, 2016 – 400 с. – ISBN 978-1-118-65720-1. – Текст : непосредственный.

11. Титтел, Э. HTML5 и CSS3 для чайников / Э. Титтел, К. Минник.  
– Москва : Вильямс, 2016 – 400 с. – ISBN 978-1-118-65720-1. – Текст :  
непосредственный.
12. Титтел, Э. HTML5 и CSS3 для чайников / Э. Титтел, К. Минник.  
– Москва : Вильямс, 2016 – 400 с. – ISBN 978-1-118-65720-1. – Текст :  
непосредственный.
13. Титтел, Э. HTML5 и CSS3 для чайников / Э. Титтел, К. Минник.  
– Москва : Вильямс, 2016 – 400 с. – ISBN 978-1-118-65720-1. – Текст :  
непосредственный.