



15. Криулин К.Н., Уткин К.Л. Кольцевой дренаж открытых плоскостных сооружений. Обоснование параметров // Неделя науки ИСИ: сборник материалов всероссийской конференции, 03-09 апреля 2023 г. В 3 ч. Ч. 1. СПб.: ПОЛИТЕХ-ПРЕСС, 2023. С. 237-239.

16. Rice M. 21 AI real estate companies to know. 2019. URL: <https://builtin.com/artificial-intelligence/ai-real-estate> (дата обращения: 09.04.2023).

17. Алексеева Т. Р. BIM-технологии и искусственный интеллект в инфраструктурном строительстве // БСТ: Бюллетень строительной техники. 2022. № 1 (1049). С. 52-54.

18. Pan Y., Zhang L. Roles of artificial intelligence in construction engineering and management: A critical review and future trends // Automation in Construction. Vol. 122, February 2021.

© К. Л. Уткин, О. Н. Уткина

**Ссылка для цитирования:**

Уткин К. Л., Уткина О. Н. Аппроксимация графических зависимостей средствами машинного обучения // Инженерно-строительный вестник Прикаспия : научно-технический журнал / Астраханский государственный архитектурно-строительный университет. Астрахань : ГАОУ АО ВО «АГАСУ», 2023. № 3 (45). С. 118-124.

УДК 004.657

DOI 10.52684/2312-3702-2023-45-3-124-128

**АДАПТАЦИЯ ПЛАГИНА CF 7 ПОД CMS WORDPRESS  
ДЛЯ РЕШЕНИЯ ЗАДАЧ РАЗВИТИЯ ТЕРРИТОРИЙ ГОРОДСКОЙ СРЕДЫ**

*Л. С. Кузякина, В. В. Соболева*

**Кузякина Людмила Семеновна**, магистрант, Астраханский государственный архитектурно-строительный университет, г. Астрахань, Российская Федерация, тел.: + 7 (964) 889-49-99; e-mail: Untitled-11@yandex.ru;

**Соболева Вера Владимировна**, кандидат педагогических наук, доцент, исполняющий обязанности заведующего кафедрой систем автоматизированного проектирования и моделирования, Астраханский государственный архитектурно-строительный университет, г. Астрахань, Российская Федерация, тел.: + 7 (905) 362-11-50; e-mail: veravsoboleva@yandex.ru

В статье представлены основные возможности использования плагина Contact Form 7 под CMS Wordpress для решения задач развития территорий городской среды и способы его адаптации. Рассмотрены вопросы, возникающие при создании системы поддержки принятия решений на основе CMS Wordpress, и варианты их решения. Приведены примеры кода для связи плагина с базой данных объектов городской среды, использования результатов запросов для генерации страниц и форм на сайте. В статье показано, что сайт с установленной системой управления контентом и плагином для создания форм может использоваться в качестве полноценной системы управления базой данных с удаленным доступом. Язык программирования PHP позволяет создавать полноценные приложения, вводить, обрабатывать и анализировать данные, а также встраивать элементы внешних онлайн-приложений: отчеты, диаграммы, виджеты и т. п.

**Ключевые слова:** CMS Wordpress, Contact Form 7, контактная форма, хук Wordpress, MySQL, управление сайтом.

**ADAPTATION OF THE CF 7 PLUG-IN CMS WORDPRESS FOR SOLVING THE PROBLEMS OF THE DEVELOPMENT OF TERRITORIES OF THE URBAN ENVIRONMENT**

*L. S. Kuzyakina, V. V. Soboleva*

**Kuzyakina Lyudmila Semenovna**, graduate student, Astrakhan State University of Architecture and Civil Engineering, Astrakhan, Russian Federation, phone: + 7 (964) 889-49-99; e-mail: Untitled-11@yandex.ru;

**Soboleva Vera Vladimirovna**, Candidate of Pedagogical Sciences, Associate Professor, Acting Head of the Department of Computer-Aided Design and Modeling Systems, Astrakhan State University of Architecture and Civil Engineering, Astrakhan, Russian Federation, phone: + 7 (905) 362-11-50; e-mail: veravsoboleva@yandex.ru

The article presents the main possibilities of using the Contact Form 7 plugin for Wordpress CMS to solve the problems of urban environment development and ways to adapt it. The issues that arise when creating a decision support system based on CMS Wordpress, and options for their solutions are considered. Code examples are connected the plugin to a database of urban environment objects, using query results to generate pages and forms on the site. The article shows that a website with an installed content management system and a plugin for creating forms can be used as a full-fledged database management system with remote access. The PHP programming language allows you to create full-fledged applications, enter, process and analyze data, as well as embed elements of external online applications: reports, charts, widgets, etc.

**Keywords:** CMS Wordpress, Contact Form 7, contact form, Wordpress hooks, MySQL, site management.

## Введение

Представление плагина Contact Form 7 его авторами как «конструктора контактных форм: простого, но гибкого» [1] полностью соответствует действительности. Изначально он заявлялся как средство создания контактных форм, которые могут быть размещены на сайтах для сбора информации о посетителях либо регистрации пользователей. Однако в реальности возможности данного плагина гораздо обширнее и позволяют использовать его как полноценную html-форму [1–3].

Плагин разработан Такаюки Миеси более 10 лет назад. Несмотря на то, что за эти годы появилось и исчезло множество других плагинов с аналогичным функционалом, Contact Form 7 по-прежнему остается самым популярным для создания контактных форм и постоянно обновляется. У плагина есть достаточно подробное описание по настройкам, но оно предназначено больше для пользователей и не раскрывает всех возможностей применения при работе с базами данных [1].

Целью данной статьи является раскрытие программных возможностей плагина Contact Form 7 при использовании инструментальных средств конструктора с последующей их адаптацией к решению задач развития территорий городской среды.

## Программные возможности плагина CF7 и примеры их использования

Реализация соответствующих информационных технологий позволяет сэкономить время на разработке дизайна и кода при создании сайта. Одним из преимуществ плагина Contact Form 7 является возможность, используя свои стили

для элементов формы, подстраивать шаблоны (рис. 1) под стиль сайта [1].

При работе над сайтом генерируется шорткод формы (рис. 1), который дает возможность вставить её в любое место страницы [1]. Плагин формирует html-код для браузера (рис. 2).

В редакторе могут быть подвергнуты изменению тэги формы, плейсхолдеры и идентификаторы полей. Плагин позволяет использовать не только те тэги, которые добавляются из меню, но и вообще любые html-тэги: ссылки, заголовки, разделители, поясняющие надписи. Это позволяет грамотно использовать место на странице, создавать удобный интерфейс и выделять отдельные поля на форме.

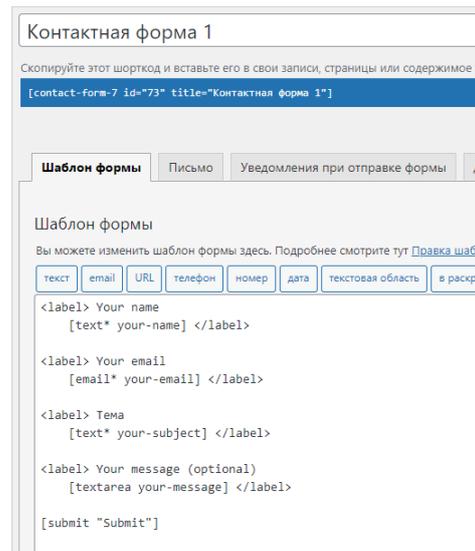


Рис. 1. Редактор форм

```
<form action=
"/straniczka-dlya-demonstracii-formy/?preview_id=201&#038;preview_nonce=c0ab1c4717&#038;p
review=true#wpcf7-f73-p201-o1" method="post" class="wpcf7-form init" novalidate=
"novalidate" data-status="init">
<div style="display: none;">
<input type="hidden" name="_wpcf7" value="73" />
<input type="hidden" name="_wpcf7_version" value="5.4.1" />
<input type="hidden" name="_wpcf7_locale" value="ru_RU" />
<input type="hidden" name="_wpcf7_unit_tag" value="wpcf7-f73-p201-o1" />
<input type="hidden" name="_wpcf7_container_post" value="201" />
<input type="hidden" name="_wpcf7_posted_data_hash" value="" />
<input type="hidden" name="wpcf7cf_hidden_group_fields" value="" />
<input type="hidden" name="wpcf7cf_hidden_groups" value="" />
<input type="hidden" name="wpcf7cf_visible_groups" value="" />
<input type="hidden" name="wpcf7cf_repeater" value="" />
<input type="hidden" name="wpcf7cf_steps" value="" />
<input type="hidden" name="wpcf7cf_options" value=
"({"form_id":73,"conditions":[],"settings":{"animations
":"yes","animation_intime":200,"animation_outtime":200,
"conditions_ui":"normal","notice_dismissed":false})" />
</div>
<p><label> Your name<br />
<span class="wpcf7-form-control-wrap your-name"><input type="text" name="your-name"
value="" size="40" class="wpcf7-form-control wpcf7-text wpcf7-validates-as-required"
aria-required="true" aria-invalid="false" /></span> </label></p>
<p><label> Your email<br />
<span class="wpcf7-form-control-wrap your-email"><input type="email" name=
"your-email" value="" size="40" class="wpcf7-form-control wpcf7-text wpcf7-email
wpcf7-validates-as-required wpcf7-validates-as-email" aria-required="true"
aria-invalid="false" /></span> </label></p>
<p><label> Тема<br />
<span class="wpcf7-form-control-wrap your-subject"><input type="text" name=
"your-subject" value="" size="40" class="wpcf7-form-control wpcf7-text
wpcf7-validates-as-required" aria-required="true" aria-invalid="false" /></span>
</label></p>
<p><label> Your message (optional)<br />
<span class="wpcf7-form-control-wrap your-message"><textarea name="your-message" cols
=40 rows=10 class="wpcf7-form-control wpcf7-textarea" aria-invalid="false"
></textarea></span> </label></p>
<p><input type="submit" value="Submit" class="wpcf7-form-control wpcf7-submit" /></p>
<div class="wpcf7-response-output" aria-hidden="true"></div></form>
```

Рис. 2. Код формы в браузере

Обращаясь к сайту, пользователь оставляет сообщение или контакт, затем информация отправляется на почту администратора. Требуется получить эти данные, обработать и сохранить их в базе данных. Возможным вариантом сохранения информации является использование плагина Flamingo, который после отправки формы сохраняет ее в единственном поле таблицы, что усложняет извлечение. Поэтому для получения информации легче перехватить её с помощью хука (рис. 3) ещё до отправки письма [2-4].

Так как во всех формах, созданных плагином, прописывается `method="post"`, при использовании данного хука получаем доступ к ассоциативному массиву `POST` и обращаемся к его элементам так же, как и в случае обычных `php/html`-форм. После выбора формы по идентификатору производится обработка извлеченных из массива данных [2-4].

Необходимость использования плагина для работы с базой данных предполагает заполнение списка выборкой из поля связанной таблицы по внешнему ключу, для чего результат запроса вносится в массив (рис. 4) [5-7].

```
// Перед отправкой письма
add_filter( 'wpcf7_before_send_mail',
'wpcf7_before_send_mail_start_function' );

function wpcf7_before_send_mail_start_function($cf7){
    $mail=$cf7->prop('mail');
    global $wpdb;
    if($mail){

        if ($cf7->id==11){ // для формы Добавить факультет
            $new_fac_name=$_POST['fac-name'];
            $new_fac_comment=$_POST['fac-comment'];
```

Рис. 3. Использование хука для получения значений полей формы

```
// Добавляем элементы в списки из БД
add_filter('wpcf7_form_tag_data_option','add_my_values_in_list',
10, 3 );

function add_my_values_in_list($n, $options, $args){
    global $wpdb;
    if (in_array('fac-list', $options)){ // Добавляем в
        список в форме факультеты
        $query_facs = 'SELECT faculty_name FROM '.$wpdb->prefix.
            'faculties WHERE 1';
        $facs = $wpdb->get_results($query_facs);

        foreach ($facs as $fac) {
            $data[]=' ' . $fac->faculty_name . ' ';
        }
        return $data;
    }
}
```

Рис. 4. Добавление в список значений из связанной таблицы

Аналогичным образом могут быть добавлены элементы в чекбокс или радиокнопку. Для добавления полей, соответствующих выполнению условия, устанавливается плагин Conditional Fields.

Contact Form 7 позволяет вставить в код формы элемент, который может быть наполнен

любым содержимым: текстом, `html`, `php`, `javascript`. Для этого создается идентификатор и заключается в квадратные скобки, как обычный тэг в форме плагина (рис. 5), код добавляется в строковую переменную и передается идентификатору [8-9].

```
// Добавляем кнопку Инструкция
add_action( 'wpcf7_init', 'form_add_instruction' );

function form_add_instruction() {
    wpcf7_add_form_tag( 'instruction', 'form_tag_instruction' );
}

function form_tag_instruction( $tag ) {
    $data = '<a href="/instrukcziya-dlya-operatora-bd/"><label
class="button button-primary wpcf7-form-control
wpcf7-submit" >Инструкция</label></a>';
    return $data;
}
```

Рис. 5. Вставка произвольного `html`-кода в форму

Благодаря тому, что имеется возможность размещения в текстовой переменной с содержи-

мым тэга еще и кода, появляется вариант создания формы с учётом значений переменных. В данном примере форма создается с учётом `GET`-

запроса со страницы, ведущей на форму. В зависимости от значения переменной в запросе, форма заполняется данными, которые пользователь сможет изменить [10–12].

При этом в раскрывающемся списке элементы соответствуют значениям связанной таблицы, а их выбор актуален для текущей записи.

Соответствующий javascript-код добавляет или убирает отображение блока в зависимости от выбора пользователя [10].

Опция «Дополнительные настройки» плагина Contact Form 7 позволяет задать видимость

форм только для зарегистрированных пользователей, настроить деморежим, отменить отправку почты и т. д. [1, 8, 9].

Один из сложных моментов при работе с Contact Form 7 – загрузка файлов. Плагин позволяет загрузить файл в форму и доставить его на указанную в форме почту, но найти файл на сайте нельзя. При загрузке файл загружается в папку со случайным именем, отправляется в письмо, затем и файл, и папка удаляются. При разрешении данной проблемы для конкретных задач развития территорий городской среды [11, 12] рассмотрены разные варианты кода, но они не работают.

```

if (isset($_GET['id'])) {
    $orderrzltid = $_GET['id'];
}
$query_rezultdls = 'SELECT r.order_id, r.equip_id, r.apart_id,
e.er_name, r.equip_quant, t.rt_name, r.rzlt_msid,
r.rzlt_msquant, r.rzlt_swdm, r.rzlt_swdmid, r.rzlt_swdmquant
FROM '.$wpdb->prefix.'order_result r LEFT JOIN '.$wpdb->prefix.'
equipment_ref e ON e.er_id=r.equip_id LEFT JOIN '.$wpdb->prefix.'
apartments a ON a.apart_id=r.apart_id LEFT JOIN '.$wpdb->prefix.'
motion_sensors m ON m.ms_id=r.rzlt_msid LEFT JOIN '.$wpdb->
prefix.'room_types t ON t.rt_id=a.room_id WHERE r.orderrzlt_id =
' . $orderrzltid . ''';
$resultdls = $wpdb->get_row($query_rezultdls);
    
```

Рис. 6. Выполнение запроса с учетом значения переменной, переданной в GET-запросе

```

$data.='><p><label> Диммер </label><br /><select
name="selectdm" class="wpcf7-form-control wpcf7-select"
aria-invalid="false">';
    foreach ($dimmers as $dimmer) {
        $data.<option value="'. $dimmer->dimmer_id . "'>
        if ($resultdls->rzlt_swdm==0) {
            $data.= ($dimmer->dimmer_id==$resultdls->rzlt_swdmid
            ? ' selected' : '';
        }
        $data.='>'. $dimmer->dimmer_name . '</option>';
    }
    $data.='</select></p>';
    
```

Рис. 7. Список заполняется значениями из связанной таблицы

```

<script>
function optSwitch() {
    var switchname = 'groupsw';
    var dimmername = 'groupdm';
    document.getElementById(switchname).style.display =
    "block";
    document.getElementById(dimmername).style.display =
    "none";
}
function optDimmer(dimmernum) {
    var switchname = 'groupsw';
    var dimmername = 'groupdm';
    document.getElementById(switchname).style.display =
    "none";
    document.getElementById(dimmername).style.display =
    "block";
}
</script>
    
```

Рис. 8. Вставка произвольного js-кода в форму

Поэтому авторами статьи принято решение воспользоваться Wordpress - CMS с открытым исходным кодом [13–15]. На диске создается папка, в которой предполагается сохранять файлы. Обращаясь к каталогу плагина с файлом /includes/file.php, отвечающем за описанные

действия, заменяем имя папки со случайного на прописанное. Данные манипуляции предоставляют возможность теперь легко найти, переместить и переименовать файл, добавить на него ссылку в базе данных.



### Заключение

Таким образом, в случаях, когда не требуется взаимодействие с базой данных и редактирование кода, целесообразно пользоваться готовыми решениями [16]. Для выполнения задач,

связанных с извлечением данных из базы, редактированием, добавлением и удалением записей, следует выбирать такие надстройки, которые можно было бы кастомизировать. Плагин для CMS Wordpress Contact Form 7 эффективно работает и в том, и в другом случае.

### Список литературы

1. Такаюки Миеси. Документация плагина Contact Form 7; URL: <https://contactform7.com/>.
2. Камаев Т. Contact Form 7 - создание и настройка форм; URL: <https://wp-kama.ru/plugin/contact-form-7>.
3. CMS Wordpress русская версия. URL: <https://ru.wordpress.org/>.
4. Кодекс Wordpress. URL: <https://codex.wordpress.org/>.
5. Кудлай А. Работаем с базой данных в WordPress. Основы работы с базой данных в WordPress, с помощью класса wpdb; URL: <https://webfor myself.com/vazhnejshie-osnovy-wordpress-vzaimodejstvie-s-bazoj-dannyx/>.
6. Рудрастых М. Работа с базой данных в WordPress. Объект \$wpdb; URL: <https://misha.agency/wordpress/interfacing-with-the-database.html>.
7. Галкин Н. Понимание и работа с данными в WordPress; URL: <https://habr.com/ru/post/233479/>.
8. Contact Form 7 настройка; URL: <https://seoblog.life/uchebnik/vordpress/contact-form-7-nastrojka.html>.
9. Полное руководство по Contact Form 7; URL: <https://contactform7.ru/>.
10. Дакетт, Дж. HTML и CSS. Разработка и создание веб-сайтов/ Дж. Дакетт- М.: Эксмо, 2019 – 480 с.
11. Прошунина, К.А., Петрова, И.Ю. Информационная система проектной деятельности в историко-культурной среде / К.А.Прошунина, И.Ю.Петрова // Инженерно-строительный вестник Прикаспия. – 2020. – № 1 (30). – С. 78–83.
12. Шиккульский, М. И., Евсина, Е. М., Кравченкова, Т. П. Разработка математической модели распределения фонда стимулирующих выплат с учетом рейтинговых оценок профессорско-преподавательского состава высшего учебного заведения / М.И.Шиккульский, Е.М.Евсина, Т.П.Кравченкова // Инженерно-строительный вестник Прикаспия. – 2022. – № 1 (39). – С. 157–163.
13. Уильямс, Б. Wordpress для профессионалов / Б.Уильямс – СПб.: Питер, 2014 – 464 с.
14. Грачев А.С. Создаем свой сайт на WordPress: быстро, легко и бесплатно/ Грачев А.С. – СПб.: Питер, 2013 – 272 с.
15. Молочков В.П., Wordpress с нуля/ Молочков В.П. – СПб.: BHV, 2021 – 304 с.
16. Зарипова В. М., Петрова И. Ю. Унаследованные информационные системы. проблемы и решения – Инженерно-строительный вестник Прикаспия. 2022. № 2 (40). С. 130–136.

© Л. С. Кузякина, В. В. Соболева

Ссылка для цитирования:

Кузякина Л. С., Соболева В. В. Адаптация плагина cf7 под cms wordpress для решения задач развития территорий городской среды // Инженерно-строительный вестник Прикаспия : научно-технический журнал / Астраханский государственный архитектурно-строительный университет. Астрахань : ГАОУ АО ВО «АГАСУ», 2023. № 3 (45). С. 124–128.

УДК 004.9

DOI 10.52684/2312-3702-2023-45-3-128-133

## РЕИНЖИНИРИНГ БИЗНЕС-ПРОЦЕССА РАЗРАБОТКИ УЧЕБНЫХ ПЛАНОВ С УЧЕТОМ КОМПЕТЕНЦИЙ И ИНДИКАТОРОВ ИХ ДОСТИЖЕНИЯ

*И. В. Аксютин, В. М. Зарипова, И. Ю. Петрова*

**Аксютин Ирина Владимировна**, доцент кафедры высшей математики и программирования, МИРЭА – Российский технологический университет, г. Москва, Российская Федерация;

**Зарипова Виктория Мадияровна**, доцент кафедры высшей математики и программирования, МИРЭА – Российский технологический университет, г. Москва, Российская Федерация;

**Петрова Ирина Юрьевна**, профессор кафедры высшей и прикладной математики, Астраханский государственный технический университет, г. Астрахань, Российская Федерация

В статье описан процесс проектирования учебных планов с учетом компетенций и индикаторов достижения компетенций для различных направлений и специальностей в университете с целью доработки подсистемы распределения компетенций в типовой конфигурации «1С: Университет ПРОФ». Разработаны BPMN диаграммы проверки качества учебного плана с точки зрения распределения компетенций, индикаторов их достижения и результатов обучения по дисциплинам учебного плана («as is» и «to be»). Разработана диаграмма DMN и составлены таблицы решений в соответствии с правилами проверки правильности распределения компетенций по дисциплинам учебного плана. Показано, что автоматизация проверки учебного плана с точки зрения распределения компетенций позволяет высвободить сотрудника учебно-методического управления УМУ от проведения такой проверки вручную.

**Ключевые слова:** компетенция, учебный план, дисциплина, контроль на согласованность, BPMN диаграмма, DMN диаграмма.