

## РОЗРОБКА САЙТА КОМЕРЦІЙНИХ ПОСЛУГ АВТОМАЙТЕРНІ СУЧАСНИМИ ЗАСОБАМИ ПРОГРАМУВАННЯ

*Анотація.* Обґрунтовано актуальність проблеми забезпечення зручності надання послуг комерційного характеру. Проаналізовано основні тенденції розвитку автомайстерень та визначено цінові категорії на послуги, що забезпечує даний інтернет-ресурс. Запропоновано методика розробки веб-проекту, що забезпечує адаптивність та кросбраузерність. Показана практична реалізація технологій Ajax з використанням написаного мовою програмування PHP скрипта, що забезпечують можливість реалізації зворотного зв'язку між користувачами веб-сайту та його адміністраторами.

**Ключові слова:** автомайстерня, веб-розробка, сайт, скрипт, функціональні можливості.

### 1 Вступ

Створення та підтримка сайтів є однією з найпопулярніших сфер сучасної веб-розробки. Дана галузь стрімко розвивається та прогресує, що зумовлює використання нових підходів та технологій. Тенденції використання різноманітних гаджетів доводять, що сучасний веб-проект повинен бути адаптованим під більшість із них, тому забезпечення адаптивності є невід'ємною складовою в сучасному сайті. Метою розробки сайтів є розширення обсягу доступності інформації, підвищення якості надання послуг. [1, 2]

В наш час актуальна проблема забезпечення зручності надання послуг комерційного характеру для кінцевого користувача. Обираючи тему сайту,

основною ціллю було повною мірою забезпечити комфортну реалізацію функцій автомайстерні.

## **2 Аналіз та тенденції розвитку сучасних автосервісів**

Більшість сучасних компаній та підприємств, як комерційного характеру так і інших видів діяльності використовують веб-сайти. Це зумовлено тим, що технології веб-розробки забезпечують широкий спектр функцій, що реалізують поставлені задачі у вигляді сайту. [3]

Автосервіс – це сукупність певних заходів, що спрямовані на задоволення потреб споживача в технічному обслуговуванні за колісними транспортними засобами, зокрема, за автомобілями. Сучасний автомобільний сервіс стрімко прогресує в різноманітних галузях своєї діяльності. Не так давно кількість майстерень автомобілів була незначною в порівнянні з сьогоденням, а їх основними завданнями було виконання найпростіших видів робіт з обслуговування та ремонту техніки, наприклад операцій зі зміни моторного масла або гальмівних колодок. Невпинне зростання автомобільного парку зумовлює збільшення виробничих потужностей, тобто зростання числа підприємств автомайстерень, підвищення рівня їхньої технічної оснащеності та професіоналізму працівників. [4, 5]

Поряд з цим відбуваються зміни у вимогах клієнтів автосервісів: вони стають більш розбірливими. У таких ситуація значну перевагу отримують ті підприємства, що не просто пропонують послуги, а й здатні забезпечити їх високу якість та продуктивність, що відповідає світовим стандартам.

## **3 Технології та підходи розробки сайту автомайстерні**

Одним із напрямів в реалізації функцій автомайстерні є сайт такого підприємства чи організації. В ньому зазначається інформація про майстерню, наприклад контактні дані, перелік послуг чи товарів, зазначення цінкових категорій та інші відомості, що бажає внести замовник. Але поряд з

інформативною частиною наповнення сайту важливу роль відіграє функціональний аспект, що включає в себе можливість здійснювати замовлення послуг або товарів безпосередньо за допомогою сайту. Така властивість значно спрощує діяльність кінцевого користувача (здійснення замовлення, отримання онлайн-консультацій) та надає гнучкий механізм управління замовленнями адміністраторам сайту.

Сучасні прогресуючі умови розвитку автосервісів вимагають від їх власників постійної модернізації та вдосконалення, зокрема і в галузі веб-технологій. Необхідно забезпечувати функціональні можливості автомайстерні та відображати ці можливості у вигляді веб-сайту. Для цього було використано ряд технологій веб-розробки, зокрема мова програмування PHP, система керування вмістом з відкритим кодом WordPress та набір технік веб-інтерфейсів AJAX. Дані технології використанні в розробці сайту автомайстерні для забезпечення таких функціональних можливостей сайту як здійснення замовлення послуги та створення динамічного запиту до сервера без перевантаження веб-сторінки.

В роботі сайту передбачена функція «Залишити заявку», що повинна забезпечити головний функціональний елемент – налаштування зворотного зв'язку між користувачами сайту та адміністраторами, наприклад (Рис.1). Для реалізації даної функції був використаний плагін Contact Form 7. Даний плагін забезпечує передачу інформації з полів форми на відповідну адресу електронної пошти та підтримує технології AJAX.

Користувач, здійснивши замовлення, в разі коректного заповнення всіх полів форми отримає відповідне повідомлення про успішність проведення операції. Якщо певне поле форми незаповнене чи заповнене некоректно, то відправка не відбудеться, а користувачеві буде запропоновано виправити помилки і повторно здійснити запит.

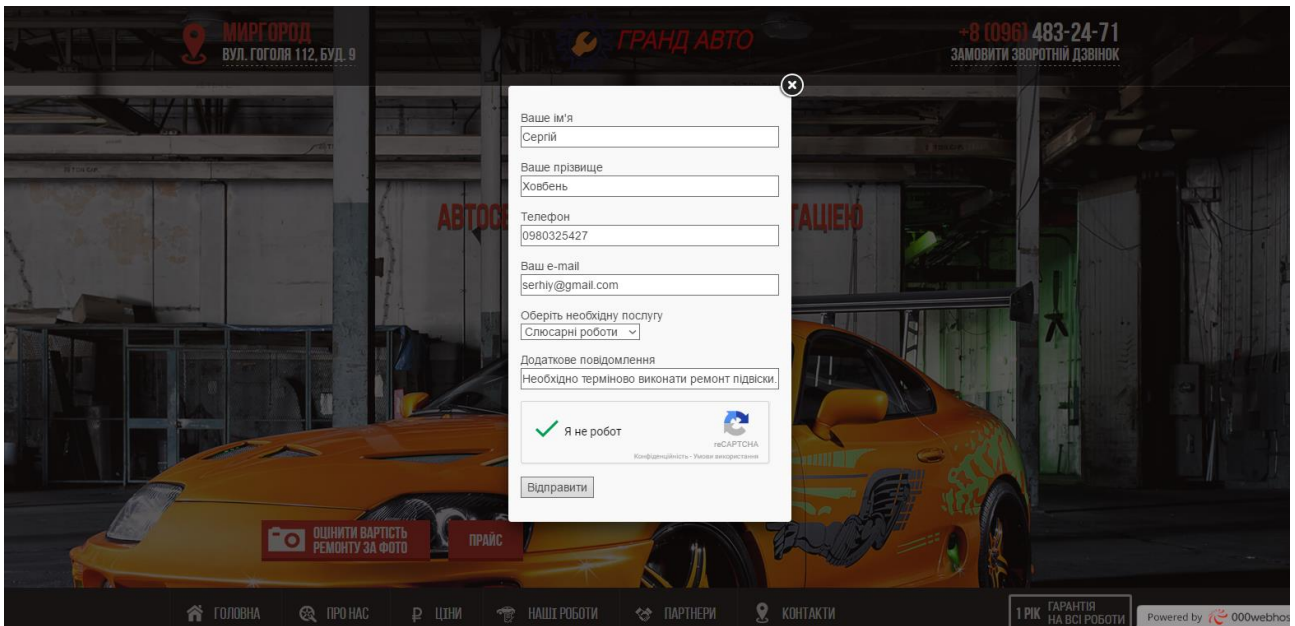


Рис.1. «Замовлення послуги»

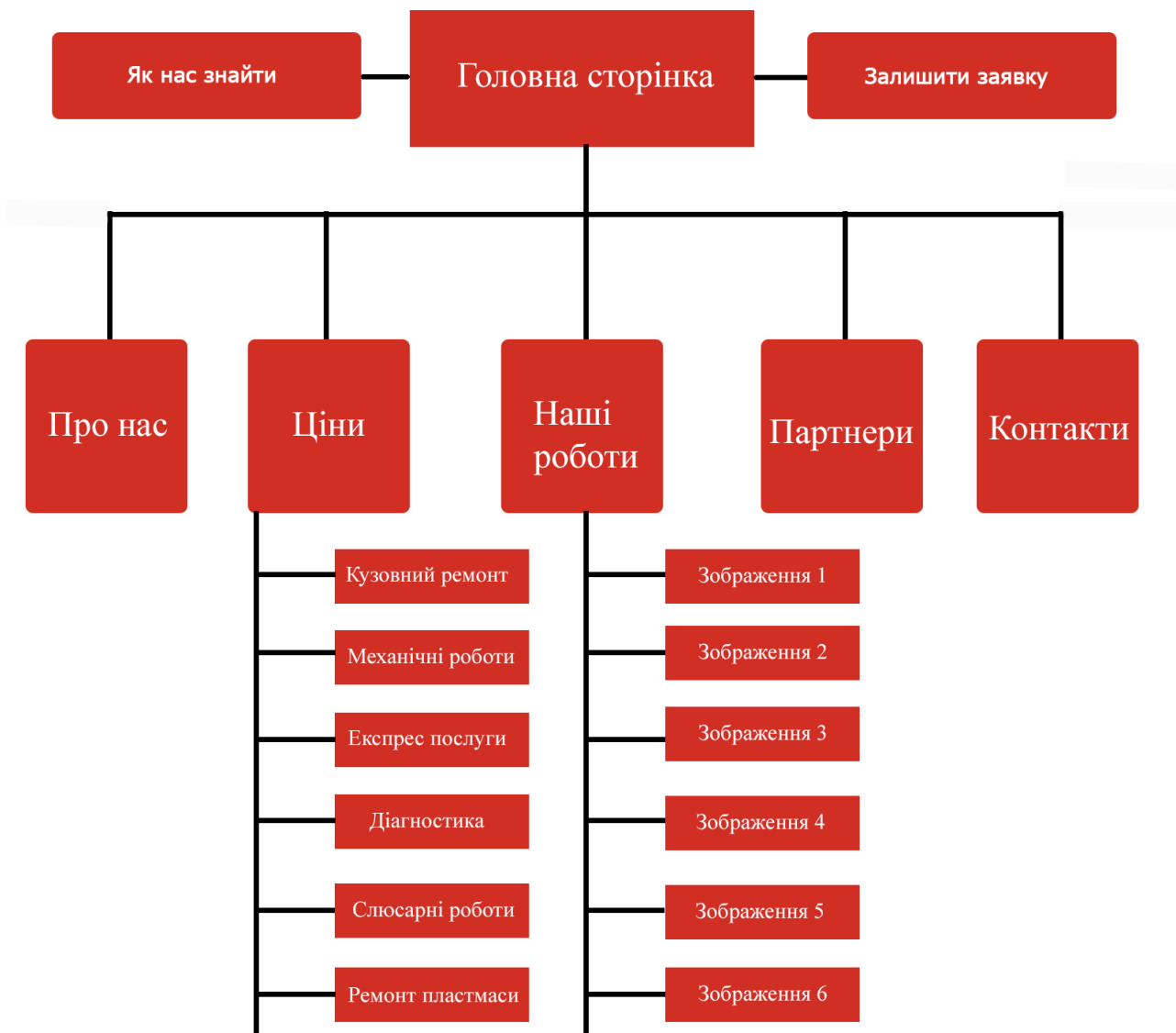
Також на сайті показано всі категорії цін для кожного з виду послуг, що забезпечує майстерня. Користувач має можливість ознайомитись з цінами на послуги, що зведені для зручності до таблиці, наприклад (Рис.2):

Ціни на послуги автомайстерні ГРАНД АВТО

Кузовний ремонт		
Рихтування кузовних деталей	1 - 2 дні	2000 грн
Фарбування	3 дні	1500 грн
Полірування	2 дні	750 грн
Механічні роботи		
Контрольно-вимірювальні роботи	1 день	500 грн
Регулювання та налаштування деталей	2 дні	800 грн
Ремонт редукторів	4 дні	1000 грн
Експрес послуги		
Комп'ютерна діагностика систем	1 день	1600 грн
Встановлення кондиціонерів	4 дні	2850 грн
Автомийка	1 день	250 грн

Рис.2 «Ціни на послуги»

Так як основним завданням сайту є спрощення, зручність та комфорт в наданні послуг кінцевому користувачеві, то внутрішня блок-схема даного веб-проекту має наступний вигляд (Рис.3):



*Рис.3 «Блок-схема сайту»*

З блок-схеми бачимо, що практичний і в той же час максимально простий перехід між сторінками сайту забезпечує зручність використання даного інтернет-ресурсу.

#### **4 Налагодження та тестування інтернет-ресурсу**

Налагодження – це процес підготовки сайту до повної його функціональності. Під час даного процесу визначаються такі важливі

компоненти як перевірка адаптивності, кросбраузерності та оптимізація сайту для пошукових систем.

У процесі створення сайту, на етапі розробки, дуже важливо перевірити працездатність сайту перед його розміщенням на хостингу. Для перевірки працездатності сайту проводиться спеціальне тестування.

Під час тестування сайт перевіряється на відповідність технічному завданню, перевіряються його технічні характеристики. Залежно від вихідного технічного завдання в процесі тестування можуть здійснюватися такі перевірки:

- перегляд сайту на різних моніторах. Перегляд сайту на моніторах різних розмірів дозволяє оцінити як буде виглядати дизайн сайту на комп'ютерах потенційних відвідувачів: чи з'являється горизонтальна смуга прокручування на маленьких екранах, чи не зміщуються елементи при різному розширенні;
- перегляд сайту в різних браузерах. Перегляд сайту в різних інтернет-браузерах і їхніх версіях, дозволяє перевірити кросбраузерність сайту;
- перевірка часу завантаження сайту на різній швидкості інтернет-підключення. Як швидко завантажуються сторінки сайту на мінімальній і на максимальній швидкості підключення;
- перевірка працездатності гіперпосилань на сайті. Чи є на сайті посилання, що ведуть не на ті сторінки. Після виявлення таких посилань їх необхідно усунути або виправити;
- перевірка коректного відображення кольорів на сайті при різних налаштуваннях колірної палітри монітора. Дозволяє довідатися, як будуть відображені кольори сайту на моніторах потенційних відвідувачів;
- перевірка відображення шрифтів, анімації й графічних зображень. Дозволяє знайти й усунути можливі проблеми із правильним відображенням зображень, анімаційних роликів і шрифтів на сайті;
- перевірка властивостей кожної сторінки сайту: заголовків, ключових слів, описи або інших мета-тегів;

- перевірка змісту кожної сторінки на відповідність вихідним матеріалам замовника й перевірка правопису на кожній сторінці. [6]

Крім того, сайт важливо протестувати на сервері. Для цього він завантажується на сервер, де багаторазово перевіряється за вищеописаними критеріями.

Окремим етапом тестування будь-якого сучасного проекту є тестування коректності коду. В даній роботі було здійснено перевірку коду за допомогою сервісу World Wide Web Consortium (W3C), який здійснює валідацію HTML та CSS файлів за такими критеріями:

- Валідація синтаксису.
- Перевірка вкладеності тегів.
- Валідація Document Type Definition (DTD).
- Перевірка на сторонні елементи.

Веб-сайт пройшов процес валідації успішно, про що свідчать результати, показані на рис.4.

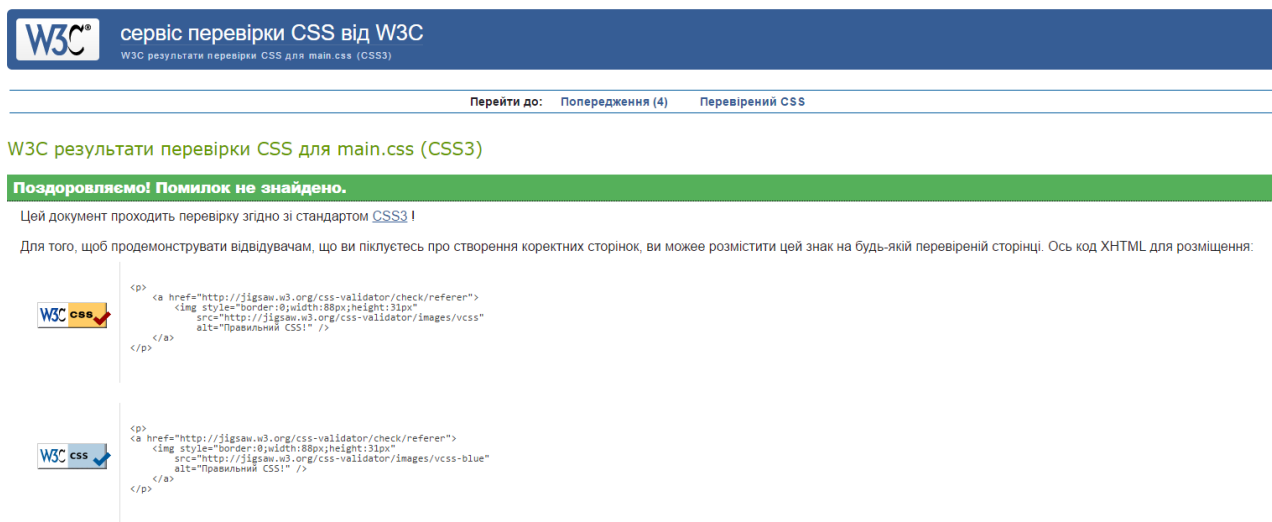


Рис.4 «Валідація файлу CSS»

## 5 Технічний опис процесу інтеграції теми сайту до системи WordPress

Для створення теми необхідно у встановленій системі WordPress створити теку з назвою шаблону. Після переходу у відповідну директорію було створено

всі необхідні файли для повноцінного функціонування ресурсу. Основними файлами для будь-якого шаблону є «index.php» та «style.css». В результаті створення даних файлів директорія має вигляд, що відображений на рис. 5.

Ім'я	Дата змінення	Тип	Розмір
css	19.05.2017 7:26	Папка файлів	
fonts	16.05.2017 11:10	Папка файлів	
img	16.05.2017 11:10	Папка файлів	
js	18.05.2017 10:48	Папка файлів	
libs	16.05.2017 16:37	Папка файлів	
contacts.php	18.05.2017 11:32	Файл PHP	9 КБ
footer.php	18.05.2017 22:58	Файл PHP	2 КБ
functions.php	18.05.2017 23:30	Файл PHP	3 КБ
header.php	19.05.2017 7:33	Файл PHP	5 КБ
index.php	19.05.2017 7:32	Файл PHP	13 КБ
mail.php	28.04.2017 17:16	Файл PHP	1 КБ
page.php	19.05.2017 7:35	Файл PHP	14 КБ
page-price.php	18.05.2017 23:06	Файл PHP	9 КБ
screenshot.png	02.03.2017 22:26	Файл PNG	1 438 КБ
sidebar.php	18.05.2017 21:16	Файл PHP	6 КБ
style.css	18.05.2017 23:46	Файл CSS	17 КБ

*Рис. 5 «Файли шаблону теми»*

Після створення усіх потрібних для роботи файлів необхідно підключити їх за допомогою спеціалізованих функцій WordPress, що написані мовою програмування PHP. Файли WordPress містять велику кількість корисних PHP функцій. Деякі з функцій, відомі як теги шаблонів, зроблені спеціально для використання в темах (шаблонах) WordPress. Також є функції, що відносяться до дій (actions) і фільтрів (filters), які використовуються в основному для розробки плагінів. Решта призначені для функціонування ядра WordPress в цілому. У кожній функції вказуються файли, в яких вона може бути знайдена (файли в директорії «wp-includes» WordPress). Усі функції WordPress за своїм призначенням поділяються на ряд груп. Розглянемо функції, що були використані для інтеграції даного проекту до системи WordPress.

Функції, що відповідають за підключення скриптів та стилів:



- `wp_enqueue_script()` – додає скрипт, тільки якщо він ще не був доданий і інші скрипти від яких він залежить зареєстровані. Залежні від даного скрипти додаються автоматично. Використовувати функцію потрібно після того як основний скрипт доданий в чергу на вивід, але ще не був виведений на екран. Тобто якщо скрипт доданий і вже виведений на екран використання `wp_add_inline_script()` нічого не виведе.
- `wp_enqueue_style()` – реєструє файл стилів, якщо він ще не був зареєстрований. Якщо файл стилів був попередньо зареєстрований через `wp_register_style()`, то його можна додати в чергу, вказавши назву (ідентифікатор) в першому параметрі `$handle`. Якщо файл не був попередньо зареєстрований, то ця функція зареєструє файл і додасть його в чергу. Доданий в чергу файл стилів виводиться в `<head>` частині документа.
- `wp_register_script()` – реєструє файл скрипта, щоб в подальшому підключати його використовуючи функцію `wp_enqueue_script()`, яка зберігає послідовність завантаження скриптів. Реєстрація скриптів потрібна, щоб взяти під контроль впровадженні скрипти: врахувати залежність одних від інших (наприклад, виводити jQuery UI скрипти після того як буде додано основний скрипт jQuery) і не виводити одні і ті ж скрипти по кілька разів. При написанні плагіна або теми рекомендується підключати скрипти цією функцією або її аналогом `wp_enqueue_script()`.
- `wp_register_style()` – реєструє CSS файл в WordPress. Після реєстрації файл можна додати в html документ за допомогою функції `wp_enqueue_style()`. Потрібно скористатися цією функцією, щоб правильно підключити CSS файли в WordPress. Для скриптів такою функцією є `wp_register_script()`. Ця функція тільки реєструє файл, щоб він був підключений і виведений, його потрібно додати в чергу за допомогою `wp_enqueue_style()`. Файл буде додано до `<head>` частини документа. Підключати файли так, а не безпосередньо в коді, вигідно

для подальшої оптимізації. Якщо всі файли підключені через систему реєстрації WordPress, то потім можна використовувати плагіни, щоб об'єднати всі стилі в один і передавати їх в стислому вигляді.

Функції, що відповідають за підключення файлів шаблону:

- `get_header()` – підключає файл шаблону `header.php` («шапку сайту»). Якщо переданий параметр `$name`, то буде підключений файл `header-{name}.php`. Якщо файл не знайдений в шаблоні теми, то буде підключено файл за замовчуванням.
- `get_sidebar()` – підключає файл шаблону `sidebar.php` («сайдбар»). Якщо переданий параметр `$name`, то буде підключений файл `sidebar-{name}.php`. Якщо файл не знайдений в шаблоні теми, то буде підключено файл за замовчуванням.
- `get_footer()` – підключає файл `footer.php` з шаблону (теми). Якщо вказано ім'я в параметрі, то буде підключений файл: `footer-{name}.php` з шаблону теми. Якщо файл не знайдений в шаблоні теми, то буде підключено файл за замовчуванням.
- `get_stylesheet_uri()` – отримує готове посилання (URL) на файл стилів `style.css` поточної теми. Файл стилів теми завжди має назву «`style.css`», тому функція використовує директиву `get_stylesheet_directory_uri()` і додає в кінці рядок «`/style.css`». В результаті отримуємо готове посилання на файл стилів. [7]

При завершенні процесу створення шаблону на панелі адміністратора були створені та підключені відповідні сторінки сайту, а також форма зворотного зв'язку. Створення сторінок сайту в системі WordPress показано на рис. 6.

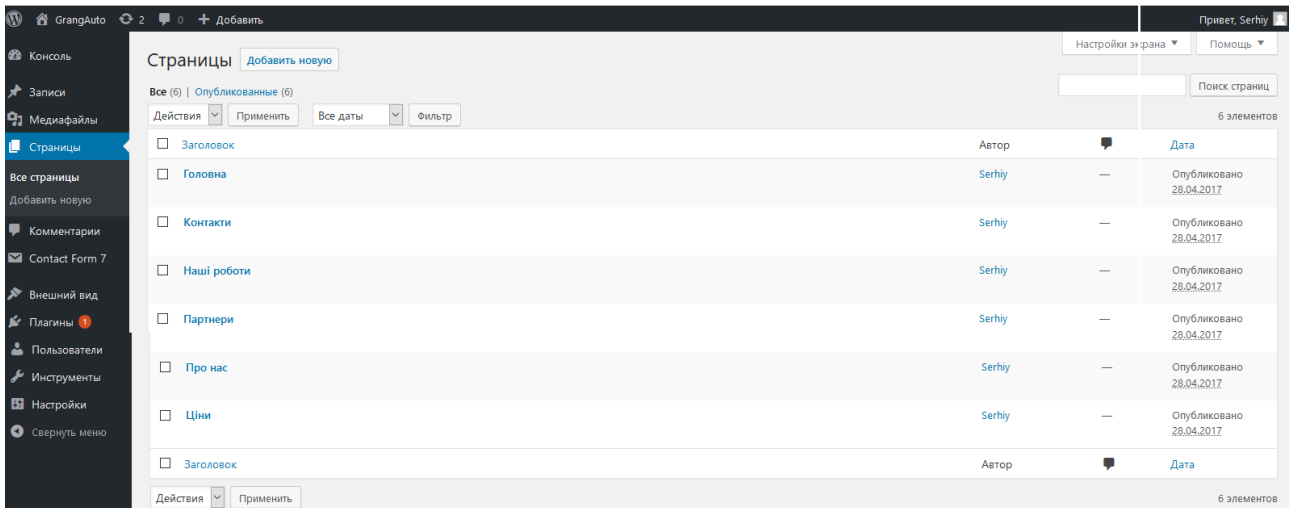


Рис. 6 «Створення сторінок сайту»

Важливим етапом налаштування сайту є перевірка його відображення при різних розширеннях екрану. Це необхідно для того щоб забезпечити адаптивність інтернет-ресурсу. Таку перевірку можна здійснити за допомогою опції «Попередній перегляд» та обрати необхідне розширення. Перевірка адаптивності сайту показана на рис. 7.

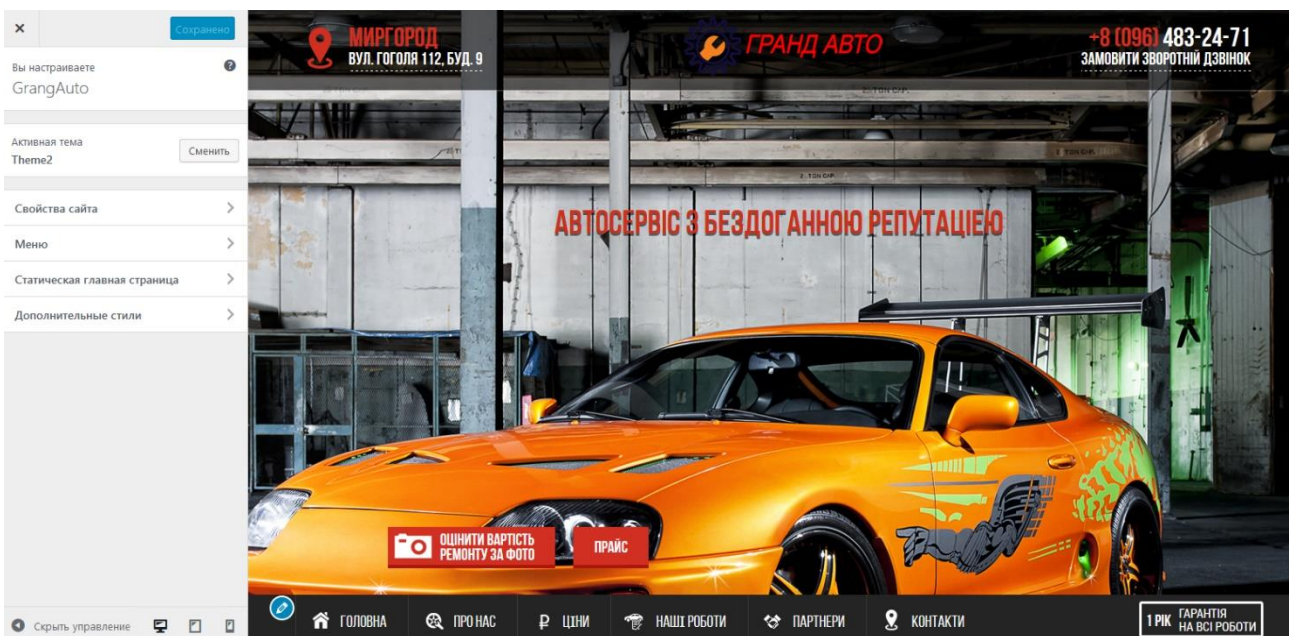


Рис. 7 «Перевірка адаптивності»

## 6 Висновок

В ході виконання роботи було здійснено аналіз мов для розробки веб-застосувань, розроблено спеціалізовані скрипти мовою програмування JavaScript, проведено налаштування функціональних можливостей сайту, виконано тестування сайту за допомогою сервісу W3C та розроблено програмний продукт, що є сайтом комерційних послуг автомайстерні «Гранд Авто».

## Посилання

1. *Пьюривал С. Основы разработки веб-приложений / С. Пьюривал; пер. з англ. О.А. Сивченко. – Санкт-Петербург: Питер, 2015. – 272с.*
2. *Маркотт И. Отзывчивый веб-дизайн / И. Маркотт; пер. з англ. П.А. Иванов, О.И. Фербер. – Москва: Символ, 2012. – 277с.*
3. *Флэнаган Д. Создание активных веб-страниц / Д. Флэнаган; пер. з англ. А.И. Киселев. – Москва: Символ, 2008. – 982с.*
4. *Системы автосервиса [Электронный ресурс] // napa.org.ua – Режим доступа до ресурсу: <http://napa.org.ua/index.php/2-uncategorised/39-sistemy-avtoservisa.html>.*
5. *Автосервіс як бізнес. Бізнес-план автосервісу [Электронный ресурс] // melnicabiz.com.ua – Режим доступа до ресурсу: [http://melnicabiz.com.ua/researches/227-avtoservis\\_kak\\_biznes-biznes-plan-avtoservisa.html](http://melnicabiz.com.ua/researches/227-avtoservis_kak_biznes-biznes-plan-avtoservisa.html).*
6. *Криспин Л. Гибкое тестирование. Практическое руководство для тестировщиков ПО и гибких команд / Л. Криспин, Д. Грегори; пер. з англ. Н.В. Мухин. – Москва: Вильямс, 2016 – 464с.*
7. *Функции [Электронный ресурс] // wp-kama.ru – Режим доступа до ресурсу: <https://wp-kama.ru/functions>.*

### Authors:

Odaruschenko E.B., Khovben S.B.

### Development of web site for car repair shop commercial services using modern programming tools

**Abstract.** Actuality problem of providing convenience to provide services of a commercial nature. The basic trends car workshop and price categories defined by the services currently provides online resources. The technique of web development project that provides adaptability and cross-

browser compatibility. Shown practical implementation using Ajax technology written in the programming language PHP script, enabling implementation of communication between users of the website and its administrators.

**Keywords:** car workshop, web development, website script functionality.

**Авторы:**

Одарущенко Е.Б., Ховбень С.В.

### **Разработка сайта коммерческих услуг автомастерской современными средствами программирования**

**Аннотация.** Обоснована актуальность проблемы обеспечения удобства предоставления услуг коммерческого характера. Проанализированы основные тенденции развития автомастерских и определены ценовые категории на услуги, что обеспечивает данный интернет-ресурс. Предложена методика разработки веб-проекта, что обеспечивает адаптивность и кроссбраузерность. Показана практическая реализация технологий Ajax с использованием написанного на языке программирования PHP скрипта, что обеспечивает возможность реализации обратной связи между пользователями сайта и его администраторами.

**Ключевые слова:** автомастерская, веб-разработка, сайт, скрипт, функциональные возможности.