

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК

ВИПУСКНА РОБОТА

**на тему: «Інформаційна система на основі алгоритму рекламного
менеджменту в мережі Інтернет з
підвищеною вимогою до інформаційної безпеки»**

Завідувач

випускаючої кафедри

Студентка гр. КБ-61

Керівник роботи

Довбиш А.С.

Киричок А.В.

Проценко О.Б.

СУМИ 2020

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК**

Затверджую _____
Зав. кафедрою Довбиш А.С.
“ _____ ” _____ 2020 р.

**ЗАВДАННЯ
до випускної роботи**

Студента четвертого курсу, групи КБ-61 спеціальності “Кібербезпека” денної форми навчання Киричок Анастасії Вадимівни.

Тема: «Інформаційна система на основі алгоритму рекламного менеджменту в мережі Інтернет з підвищеною вимогою до інформаційної безпеки»

Затверджена наказом по СумДУ
№ _____ від _____ 2020 р.

Зміст пояснювальної записки: 1) Інформаційний огляд; 2) Постановка задачі й формування завдань дослідження; 3) Вибір методу рішення задачі; 4) Практична реалізація; 5) Аналіз результатів.

Дата видачі завдання “ _____ ” _____ 2020 р.
Керівник випускної роботи _____ Проценко О.Б.
Завдання прийняв до виконання _____ Киричок А.В.

РЕФЕРАТ

Записка: 80 стор., 11 рис., 4 додатки, 16 джерел.

Об'єкт дослідження — Веб-додаток у вигляді рекламного плагіну для фреймворку WordPress.

Мета роботи — розробити веб-додаток у вигляді плагіну для фреймворку WordPress. Плагін повинен бути реалізованим для подальшого легкого та простого користування.

Результати — актуальність обраної теми для реалізації проаналізовано, спланована методологія вирішення поставленого завдання. Розроблений веб-додаток реалізовано у формі плагіну для фреймворку WordPress, програмно створеного за допомогою HTML, CSS, PHP, JS.

ВЕБ-ДОДАТОК, ПЛАГІН, ФРЕЙМВОРК, ІНФОРМАЦІЙНА
СИСТЕМА, РЕКЛАМНИЙ МЕНЕДЖМЕНТ, БЕЗПЕКА ВЕБ-
РЕСУРСУ, HTML, CSS, PHP, JS.

ЗМІСТ

ВСТУП.....	5
1. ІНФОРМАЦІЙНИЙ ОГЛЯД	6
1.1 SEO оптимізація web-сайтів	6
1.2 Рекламна діяльність в мережі Інтернет	8
1.3. Огляд подібних рішень	10
1.4 Постановка задачі.....	20
2. ВИБІР МЕТОДУ РІШЕННЯ	21
2.1 Вибір програмної реалізації.....	21
2.2. Вибір системи управління базою даних (СУБД).....	24
3. ПРАКТИЧНА РЕАЛІЗАЦІЯ.....	25
3.1 Розробка моделі інформаційної системи	25
3.2 Програмна реалізація	27
3.3 Безпека web-ресурсу	33
ВИСНОВКИ	36
СПИСОК ЛІТЕРАТУРИ.....	37
ДОДАТКИ.....	39
ДОДАТОК А	39
ДОДАТОК Б.....	46
ДОДАТОК В	59
ДОДАТОК Г.....	69

ВСТУП

Реклама – вид поширення інформації товарів чи послуг в мережі Інтернет з метою привертання уваги потенційного споживача.

Інтернет, як невід’ємлива частина життя людини, створює не тільки простір для розваг, а є своєрідним майданчиком для ведення бізнесу. Безліч інтернет-магазинів існує зараз. Усі вони надають певні послуги кінцевому користувачу. Не виходячи з дому, є можливість придбання необхідного товару в один клік. Все, що потрібно – вийти в мережу Інтернет та обрати бажане. Навіть, якщо товар не буде придбано, данні про компанію залишаться в пам’яті користувача.

Проте є один нюанс, це виключення стосується розповсюдження інформації, а саме оголошень. В цьому випадку на допомогу стає в нагоду реклама. Вона (реклама) має безліч варіантів реалізації як в онлайн середовищі, так і офлайн.

Нерідко можна зустріти інформаційні повідомлення в друкованому вигляді, на дошках оголошень, в людних місцях. Ефективності та зворотнього зв’язку із такого представлення майже немає.

Розглянемо інший вид реклами – реклама в інтернеті. Відображення даних може набувати різноманітних форм. Від банерів із привабливими слоганами до коротеньких відео із захоплюючими історіями бренду. Прочитане або побачене не залишить байдужим користувача будь-якої вікової категорії, що є головною метою рекламодавця.

Аспект, на який неодмінно слід звернути увагу, крім оригінальності текстового наповнення, це те, що усі види реклами нерідко використовуються шахраями у корисливих цілях.

Головна проблематика роботи полягає не тільки у створенні ефективного коду для просування в мережі Інтернет, а також безпечного (коду). Для унеможливлення інтеграції сторонніх скриптів із небезпечними посиланнями. Адже, в наш час – захищеність інформаційних даних є однією із головних проблем в сучасному кіберпросторі.

1. ІНФОРМАЦІЙНИЙ ОГЛЯД

1.1 SEO оптимізація web-сайтів

Пошукова оптимізація сайту або ж SEO — комплекс заходів, що націлені на збільшення рейтингу сайту в пошукових системах по цільовим запитам. Це загальне представлення оптимізації. Більш розгорнуте формулювання можна пояснити наступним чином.

Пошукова оптимізація сайту або ж SEO — комплекс заходів для розвитку сайту, зміни його оточення, аналізу поведінки користувачів, направлений для поліпшення вже існуючих позицій по цільовим запитам.

Від загальної кількості та популярності запитів, що просувається, залежить відвідування сайту із пошуку та його популярність. Проте не тільки запити, але й реклама є частиною seo оптимізації.

Методи оптимізації можна розділити на три класи – біла, сіра та чорна. Отже, почнемо розгляд кожної з них детальніше.

Біла оптимізація — комплекс робіт над сайтом, тобто над його наповненням, навігацією, а також роботи із просуванням цього сайту шляхом оглядів, партнерських програм без впливу на пошукові алгоритми сайтів.

Сіра оптимізація — комплекс робіт, котрі полягають в додаванні великої кількості ключових слів в текст сторінки, часто з втратою логічного сенсу для користувача.

Чорна оптимізація — комплекс робіт, які суперечать правилам пошукових систем. Серед них можна виділити наступні: використання дорвеїв (сторінок і ресурсів, створених спеціально для роботів пошукових систем, найчастіше з великою кількістю ключових слів на сторінці), прийом під назвою клоакінг (користувачеві віддається одна сторінка, що легко читається, а пошуковому роботу — інша, оптимізована під які-небудь запити), використання прихованого тексту на сторінках сайту, використання «однопксельних посилань».

Для того, щоб відбулося визначення рейтингу сайту існують наступні метрики.

Google PageRank — один з методів для визначення важливості сторінки, що використовує Google. Релевантні сторінки отримують більший PageRank, тобто вони найбільш часто показуються у перших результатах пошуку.

PageRank — сімейство алгоритмів оцінки важливості веб-сторінок за допомогою розв'язання систем лінійних рівнянь. Для кожної сторінки обчислює дійсне число, чим більше число — тим «важливіша» сторінка

Google PageRank (PR) — це міра від 0 до 10. Google Pagerank базується на зворотних посиланнях. PageRank працює, підраховуючи кількість та якість посилань на сторінку, щоб визначити приблизну оцінку того, наскільки важливий веб-сайт.

Alexa Traffic Rank — метод, що базується на кількості трафіку, зареєстрованого користувачами. Тобто ті юзери, що прозвели встановлення панелі інструментів Alexa протягом трьох місяців. Рейтинги на сайті базуються на загальній кількості унікальних відвідувачів та переглядів сторінок. Виняткові відвідувачі визначаються кількістю унікальних користувачів Alexa, що здійснюють вхід на сайт у певний день.

Перегляди сторінок — це загальна кількість запитів URL-адрес користувачів Alexa для сайту.

Moz Domain Authority. Розробка компанії MOZ, що являє собою метрику веб-сайту. Слугує показником трафіку веб-сайту та рейтингу.

Кількість пов'язаних кореневих доменів, кількість загальних зворотних зв'язків та відстані зворотних посилань на домашній сторінці веб-сайтів — це ті показники, на які спирається адміністрація домену.

Moz Page Authority. Алгоритм, що визначає вагу сторінки, взятої окремо. Оцінка являє собою логарифмічну шкалу на 100 пунктів. Метод розроблений Мозом. Спам не враховується на відміну від TrustFlow.

1.2 Рекламна діяльність в мережі Інтернет

Рекламна діяльність в Інтернеті - один з інструментів, який може поєднувати інші види реклами та взаємодіяти у налагоджені ділових відносин у сфері маркетингу.

Показ рекламного повідомлення інтернет-користувачу є головним завданням маркетингу мережі Інтернет, що відбувається прямим шляхом комунікації рекламодавця з відвідувачем без посередників. Це відбувається, завдяки переходу на сайт рекламованого товару чи послуги, де представлена вичерпна інформація. У цьому полягає основна відмінність інтернет-реклами від всіх інших традиційних способів рекламування.

Рекламна веб-діяльність включає в себе не тільки відображення рекламних банерів, а й просування веб-ресурсів і тд. Більш детально розглянемо рекламне функціонування у мережі Інтернет.

Просування сайту — комплекс дій для забезпечення високих позицій ресурсу в пошукових системах з метою підвищення відвідуваності сайту цільовою аудиторією. Цільова аудиторія — це зацікавлена група людей у придбанні продукції, товарів або послуг, представлених на сайті який просувається.

Одним з найдієвіших методів розкрутки сайту в пошукових системах є оптимізація. Пошукова оптимізація — це комплексна робота над зовнішніми і внутрішніми факторами сайту, з урахуванням вимог алгоритмів пошукових систем, для виведення сайту на більш високі позиції в пошуковій видачі. Таким чином збільшується цільову відвідуваність ресурсу і його конверсія.

Просування сайтів в мережі Інтернет є популярною практикою сьогодення, насамперед це необхідно для підвищення популяризації та продажу товарів чи послуг.

Електронна комерція — одна із сфер цифрової економіки, що охоплює всі транзакції, пов'язані с торгівлею та фінансами, що реалізуються на базі комп'ютерних мереж.

Цей вид комерції зазвичай використовують при користуванні онлайн магазинами. Існують такі види торгівельних майданчиків:

Інтернет-магазин — майданчик на просторах всесвітньої мережі, де реалізовується продаж товарів юридичній чи фізичній особі, враховуючи доставку. При цьому розміщення споживацької інформації, замовлення товару і угода відбуваються на сайті інтернет-магазину.

Електронний магазин — сайт, з якого можна вибрати та замовити потрібний товар чи послугу. Інтернет-магазин перш за все передбачає грошові розрахунки на відміну від сайтів, які надають послуги безкоштовно. Для безпечного використання сайту передбачені надійні протоколи — https, та інші системи безпеки.

Шопінг в Інтернеті є популярним та важливим інструментом у реальному житті. Адже, не завжди асортимент товарів чи послуг в офлайн-магазині є задовільним для користувача. Так, було розглянуто матеріальний аспект проблеми Інтернет маркетингу.

Кожного дня тисячі юзерів користуються Інтернет ресурсами задля удосконалення своїх знань та вмінь. Користувачі читають новини або різні жанрі літератури, переглядають фільми, спілкуються в різних соціальних мережах і все це має неабиякий розвиток завдяки статистиці.

Інтернет-статистика — сервіси, що дозволяють збирати і аналізувати інформацію про відвідувачів сайтів. Інформація вибирається з даних, наданих мережевими протоколами. Для отримання даних протоколу з запиту користувача вимагається, щоб разом зі сторінкою сайту виконався серверний скрипт системи інтернет-статистики. Найпростішим і найкориснішим способом зробити це є розміщення на сторінці лічильника відвідувань.

Лічильник відвідувань допомагає розробнику краще розуміти, яка інформація користується попитом у користувачів. Знаючи, цю інформацію сайт покращує та корегує своє наповнення під попит потенційного відвідувача. Окрім,

лічильника відвідувань існують такі сервіси, що відображають вік цільової аудиторії, місце знаходження, кількість цитувань посту в мережі Інтернет.

Всі ці фактори впливають на популярність та корисність ресурсів на ринку попиту веб-відвідувачів. Саме статистика сайту чи веб-сторінки грає важливу роль при виборі розміщення реклами та її ефективності.

1.3. Огляд подібних рішень

В наш час відомо велика кількість способів представлення реклами в мережі Інтернет. Нижче описано основні види інформаційних оголошень.

Медійна реклама. Вид рекламних оголошень у вигляді медіо-графічних матеріалів в мережі Інтернет, що розміщуються на спеціальних сайтах-майданчиках. Сайти-майданчики — це пошукові системи, тематичні сайти, новинні та інформаційні ресурси, — іншими словами, ті сайти, на яких можна розраховувати на свою цільову аудиторію. Здійснюючи медіапланування, обов'язково проводиться якісна оцінка аудиторії сайту-майданчика, вивчається географія відвідувачів, час проведений ними на сайті, джерела трафіку майданчики — тобто, відбувається комплексне сканування сайтів-майданчиків для виявлення потенційних клієнтів.

Медійна реклама є ефективним інструментом для підвищення популярності бренду. Такий вид оголошень відноситься до персоналізованої реклами.

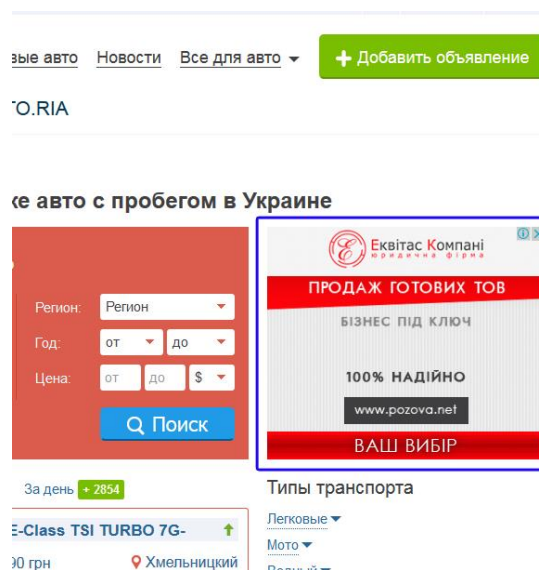


Рис.1.1. – Приклад медійної реклами в мережі Інтернет

Google AdWords — онлайн-рекламний сервіс, який дозволяє рекламодавцям конкурувати для відображення коротких рекламних текстів для веб-користувачів, частково ґрунтуючись на ключові слова, зумовлені рекламодавцями, що може зв'язати, скопіювати вміст веб-сторінок, що відображаються користувачам.

Інтернет-реклама в Google Ads ділиться на кілька видів:

1. Контекстна реклама в інтернеті.
2. Класичний та динамічний ремаркетинг.
3. Медійна (банерна) реклама в інтернеті.
4. Інтернет-реклама Google Shopping.
5. Відеореклама в інтернеті.

Персоналізована реклама – оголошення, які показуються відповідно до історичних даних або попередньо зібраної інформації про пошукові запити користувача, його активність, відвідані сайти та використовувані додатки. Також до такої інформації належать демографічні дані та відомості про місцезположення користувача.

Розміщення медійної реклами може здійснюватися за допомогою технології запропонованої компанією Google.

За допомогою технології Google AdWords рекламне оголошення можна націлити на визначену мовно або географічно аудиторію. Таке націлювання є ефективнішим для товару чи послуги, що рекламується. Оскільки, медійна реклама є персоналізованим видом реклами, то пропонується наступне рішення. Нижче наведено код персоналізованої реклами Google AdWords.

```
<html>
<head>
<title>Назва                вашого                сайту                </title>
</head>
<body>

<script async src="https://pagead2.googleadsyndication.com/pagead/js/adsbygoogle.js">
```

```

</script>
<script>(adsbygoogle=window.adsbygoogle||[]).pauseAdRequests=1;
</script>
// This will pause ad requests, so users have time to interact with your consent solution.

/* Set up the consent solution and act according to the user choice.

If the user declines personalized ads, make sure to call (adsbygoogle=window.adsbygoogle||[]).request
NonPersonalizedAds=1;
Remember that you need user consent for using cookies even for non-
personalized ads in countries where the EU eprivacy directive requires it.

Later, you can call (adsbygoogle=window.adsbygoogle||[]).pauseAdRequests=0 to resume sending ad r
equests. Without making this call, no ads will be shown.

*/

<!-- One test unit for GDPR -->
<ins class="adsbygoogle" style="display:inline-block;width:970px;height:250px"
  data-ad-client="ca-pubxxx"
  data-ad-slot="slot_id">
</ins>

<!-- Another test unit for GDPR -->
<ins class="adsbygoogle" style="display:inline-block;width:250px;height:250px"
  data-ad-client="ca-pubxxx"
  data-ad-slot="slot_id">
</ins>
<script>(adsbygoogle = window.adsbygoogle || []).push({});</script>
// This usually triggers the ad request, but you have paused these.

</body>
</html>

```

Автоматизовані оголошення - угруповання інформаційних оголошень в мережі Інтернет, які дозволяють реалізувати монетизацію примітивним способом. Такий вид оголошень з'являється автоматично при найбільш необхідних умовах, не заважаючи споживачу.

Переваги автоматизованої реклами:

- прості у використанні;
- квінтесенція оголошення у його оригінальному змісті;
- можуть вплинути на примноження доходу.

Принцип роботи автоматизованих оголошень полягає у наступному. Проводиться аналіз розміщення елементів на сторінці та наявна реклама на ній, виходячи з цих даних, рекламні оголошення розміщаються автоматично. Нижче наведена програмна реалізація автоматизованого оголошення, а саме - код для персоналізованої реклами.

```
<html>
<head>
<title>Название                вашего                сайта</title>
<script async src="https://pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></script>
<script>
(adsbygoogle=window.adsbygoogle||[]).pauseAdRequests=1;
// This will pause ad requests, so users have time to interact with your consent solution.
(adsbygoogle=window.adsbygoogle||[]).push({ google_ad_client: "ca-
pubxxx", enable_page_level_ads: true });
// This usually triggers the ad request, but you have paused these.
</script>
</head>
<body>

<script>
/* Set up the consent solution and act according to the user choice.

If the user declines personalized ads, make sure to call (adsbygoogle=window.adsbygoogle||[]).request
NonPersonalizedAds=1;
```

Remember that you need user consent for using cookies even for non-personalized ads in countries where the EU eprivacy directive requires it.

Later, you can call `(adsbygoogle=window.adsbygoogle||[]).pauseAdRequests=0` to resume sending ad requests. Without making this call, no ads will be shown.

```
*/
</script>

</body>
</html>
```

Неперсоналізовані оголошення. Неperсоналізованою вважається та реклама, що показується без урахування попередньої активності користувача. У такому разі націлювання здійснюється на основі контекстуальної інформації: враховуються поточні пошукові запити користувача або вміст сайту чи додатка, що зараз використовується, а також застосовується загальне географічне націлювання (на рівні міста) відповідно до поточного місцеположення особи. Для цього типу реклами Google забороняє використовувати будь-які типи персоналізованого націлювання, зокрема демографічне та націлювання за списком користувачів.

У такій рекламі файли cookie й ідентифікатори оголошень для мобільних пристроїв не використовуються саме для націлювання, однак вони застосовуються для обмеження частоти показів, створення зведених звітів про оголошення, а також для протидії шахрайству та порушенням. Тому, відповідно до Директиви про конфіденційність і електронні засоби зв'язку, для деяких країн Європейської економічної зони необхідно отримати згоду на використання файлів cookie й ідентифікаторів оголошень для мобільних пристроїв (якщо це передбачено законодавством). Приклад коду для неперсоналізованого оголошення:

```
<html>
<head>
<title>Заголовок вашого сайту</title>
```

```

<script async src="https://pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></script>
<script>
(adsbygoogle=window.adsbygoogle||[]).pauseAdRequests=1;
// Рядок, що призупинить запити оголошень на час.
(adsbygoogle=window.adsbygoogle||[]).push({ google_ad_client: "ca-
pubxxx", enable_page_level_ads: true });
// Рядок, що зазвичай активує запит оголошення, однак вони призупинені.
</script>
</head>
<body>

<script>

```

Банерна реклама. Реклама, яка є найпоширенішою із усіх видів оголошень у Всесвітній мережі, яка презентує графічний блок, наповнення якого може містити різноманітні представлення інформації. Як у вигляді зображень із текстовим наповненням, так і відео матеріали.

Класичним прикладом банерної реклами є прямокутник із графічним зображенням (в форматі .jpeg, .gif, .png) або програмно реалізований банер за допомогою технології JavaScript, PHP тощо. Завдяки кодуванню оголошення може приймати будь-яку геометричну форму, а також мати аудіо та відео супровід.

Банерні оголошення – потужний інструмент в іміджевій рекламі бренду або компанії. Проте не завжди такий вид оголошень є посиланням на корисний сайт. Залучення вірусних посилань у програмний код стало звичайною справою Інтернет-шахраїв сьогодення. Схема роботи такого переходу наступна: відкривається сайт із сумнівною репутацією та відбувається завантаження файлів. Ці документи є вірусами, які порушують роботу технічного забезпечення користувача. Щоб зменшити вірогідність потрапляння такого контенту потрібно користуватися послугами спеціальних баннерно-обмінних майданчиків чи впроваджувати через безпосередній контакт із адміністратором сайту.

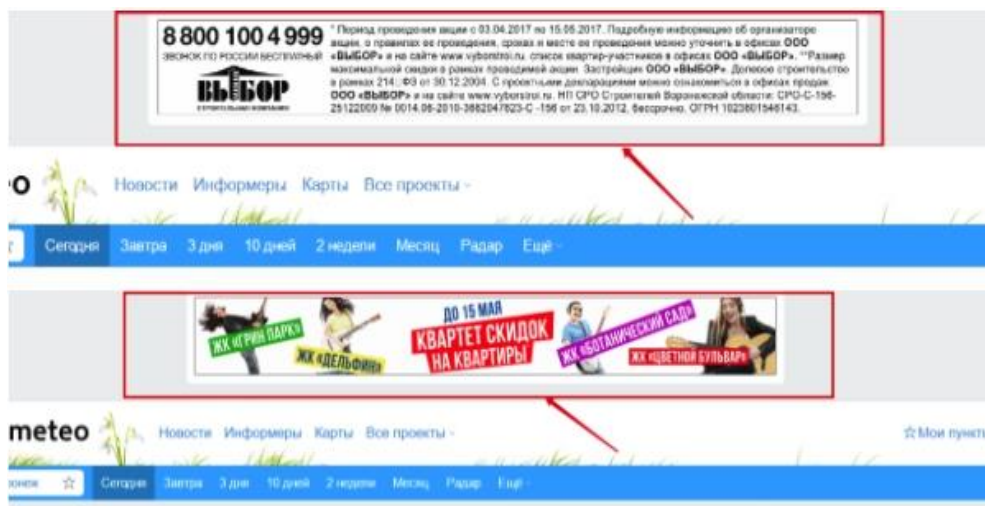


Рис.1.2 – Приклад банерної реклами в мережі Інтернет

Контекстна реклама. Найуспішніший вид реклами в мережі Інтернет. Вибір рекламних оголошень на основі інтересів конкретного користувача. Такий підбір реалізується за допомогою аналізу запитів, відвідуваних сайтів.

Перевагою данного виду реклами є геотаргетинг, можливість вибору географічного показу сторінок. Також в контекстній рекламі наявні обмеження за часом показу.

Принцип роботи контекстної реклами полягає у пошуку при введенні ключового слова. Оголошення контекстної реклами в інтернеті показуються в пошуковому рядку, як наведено на рисунку нижче:

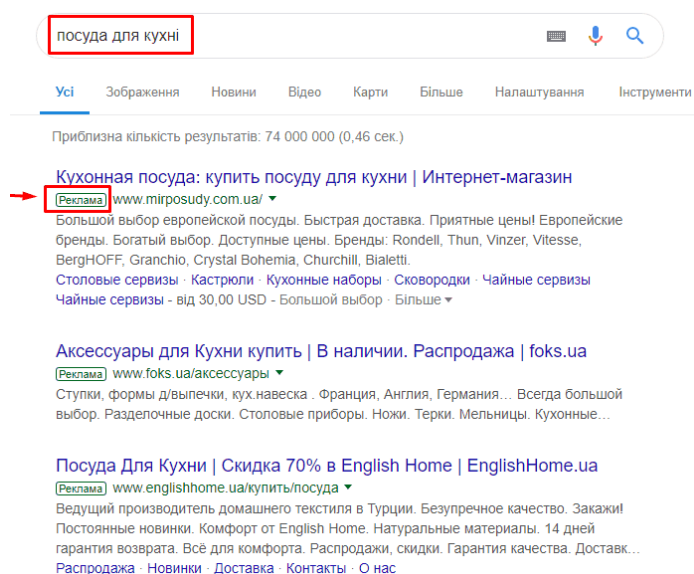


Рис. 1.3 – Приклад контекстної реклами в мережі Інтернет

Такий вид рекламних оголошень у мережі Інтернет є найбільш легальним та безпечним. Користувачі отримують бажаний результат у виді набору посилань, які є доречними в даному контексті пошуку.

Провайдерами контекстної реклами є самі пошукові системи. Найбільші сервіси такого типу — Google AdWords і Яндекс.Директ.

Яндекс. Директ — система контекстної реклами на сторінках Яндекса і на сайтах-учасниках рекламної мережі Яндекса. На Яндексі також доступні й інші види контекстної реклами, наприклад, медійний контекстний банер чи Яндекс.Маркет, які можна використовувати одночасно з Директом.

Можливості, які надає даний рекламний сервіс:

- перегляд статистики пошукових запитів за місяць, як загальної так і по регіонам світу;
- вибір запитів, при котрих показуватиметься рекламне оголошення;
- можливість скласти рекламний текст;
- геотаргетинг — можливість вибрати країну та/або місто, де мешкає потенційний покупець;
- керувати своєю рекламою на сторінках результатів пошуку Яндекса.

Розглянемо принцип роботи сервісу Яндекс.Директ. Нижче приведено приклад асинхронного коду Яндекс Директ.

```
<!-- Yandex.Direct D-A-128107-27 -->
<div id="yandex_direct_D-A-128107-27"></div>
<script type="text/javascript">
  (function(w, d, n, s, t)
  {
    w[n] = w[n] || [];
    w[n].push(function()
    {
      Ya.Context.AdvManager.render({
        blockId: "D-A-128107-27",
        renderTo: "yandex_direct_D-A-128107-27"
      });
    });
  });
```

```

};
t = d.getElementsByTagName("script")[0];
s = d.createElement("script");
s.type = "text/javascript";
s.src = "//an.yandex.ru/system/context.js";
s.async = true;
t.parentNode.insertBefore(s, t);
}
(this, this.document, "yandexContextAsyncCallbacks");
</script>

```

На рисунку зображеному нижче, приведено приклад додавання банера с кодом Яндекс Директ з partner2.yandex.ru.

Содержимое баннера

Код Яндекс Директ partner2.yandex.ru [Sync_Async]

Параметры

Основные параметры

Название	Код Яндекс Директ partner2.yandex.ru
Приоритет	100
Статус	<активен>
Отключен на новых площадках	<input checked="" type="checkbox"/>
Профиль таргетирования	<не переопределен>
События включены	<input type="checkbox"/>

Содержимое

renderTo, до части с blockID	yandex_direct
blockID	D-A-128107-27

Общие ограничения

Показы всего, сутки, час	
Переходы всего, сутки, час	
Дата, время начала	17.11.16 00:00
Дата, время окончания	23:59

+ Добавить

Рис. 1.4 - Банер с кодом Яндекс Директ з partner2.yandex.ru

Код Яндекса буде виконаний на сторінці сайту (не в iframe adfox).

Проте, варто зазначити, що даний сервіс є заблокований з червня 2017 року для користувачів, які проживають на території України.

Інтернет-реклама в соціальних мережах. Виникнення та розповсюдження соціальних мереж стало неабияким поштовхом для глобалізації населення. Сучасний індивід вже не уявляє свого життя без звичних соціальних мереж, де вирує своє життя. Безлімітний перегляд фільмів, оплата комунальних платежів, безкоштовний дзвінок у будь-яку точку світу стали буденністю. Навіть реклама стала більш гнучкою та модернізованою.

Реклама в соціальних мережах є ефективною та таргетованою, тобто націленою на зацікавленого користувача.

В соціальних мережах існує угруповання навколо сторінки бренду, де збирається зацікавлена аудиторія. В Україні найбільш поширеними каналами для реклами є Facebook та Instagram.

Наприклад, такий вигляд має таргетована реклама у Facebook:

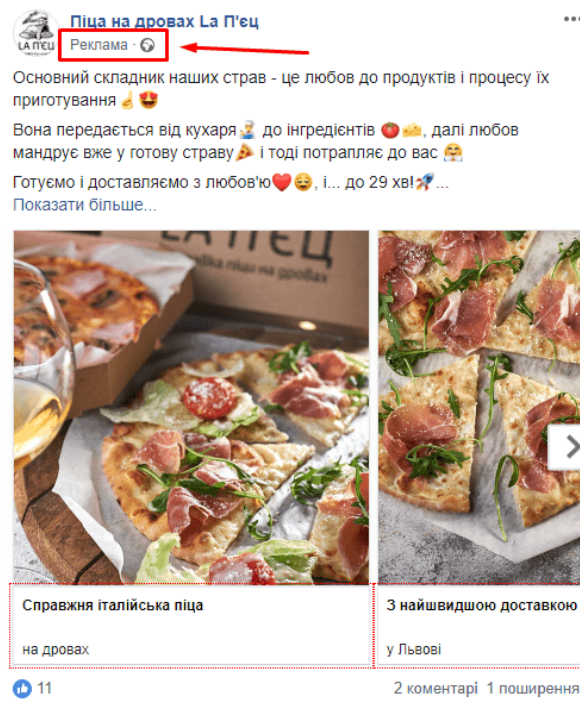


Рис. 1.5 - Таргетована реклама у Facebook

Та таргетована реклама в Instagram:

Це один із тривіальних та бюджетних способів запуску реклами. Навіть приватна особа, яка представляє свої послуги певній територіальній місцевості, може замовити таку рекламу, обравши розмір охоплення аудиторії, вікові

особливості та геолокацію. Для впровадження оголошень на соціальних каналах не потрібно кодування. Обирається лише рекламний матеріал.

Вірусна реклама. Вид реклами у мережі Інтернет, яке є предметом розповсюдження цільової аудиторії даної ідеї. Рекламні оголошення з такою тематикою побудовані на довірчій або шокуючій інформації.

Мета - створення вірусного наповнення, яке буде поширене користувачами мережі Інтернет. Прикладом таких рекламних оголошень можуть виступати flash-додатки, інтригуючі відеоролики або текст.

Вірусна реклама може нести не тільки інформаційний непередбачуваний контент, але і реальні комп'ютерні віруси, що буквально знищують операційну систему користувача. Так під виглядом завантаженої статті із актуальними новинами можуть бути трояни.

1.4 Постановка задачі

Необхідно розробити плагін для рекламних оголошень та інтегрувати його у фреймворк Wordpress. Слід реалізувати візуальну та серверну частину (Front-end та Back-end) веб-плагіну, із таким функціоналом.

Клієнт:

1. Додавання оголошень у графічному та текстовому вигляді;
2. Перегляд оголошень;
3. Налаштування зовнішніх факторів відображення оголошень;
4. Вибір персональних налаштувань для відображення рекламних оголошень.

Адміністратор:

1. Всі можливості клієнта;
2. Доступ до панелі адміністратора та всі її можливості.

Сервер:

1. Доступ до бази даних, можливість вносити зміни;
2. Взаємодія з шаблонами веб-сайту.

2. ВИБІР МЕТОДУ РІШЕННЯ

2.1 Вибір програмної реалізації

Фреймворк — інфраструктура програмних рішень, що полегшує розробку складних систем. Спрощено дану інфраструктуру можна вважати своєрідною комплексною бібліотекою, але при цьому вона має ряд обмежень, що задають правила створення структури проекту та написання коду.

Програмний фреймворк — це готовий до використання комплекс програмних рішень, включаючи дизайн, логіку та базову функціональність системи або підсистеми. Відповідно — програмний фреймворк може містити в собі також допоміжні програми, деякі бібліотеки коду, скрипти та загалом все, що полегшує створення та поєднання різних компонентів великого програмного забезпечення чи швидке створення готового і не обов'язково об'ємного програмного продукту. Побудова кінцевого продукту відбувається, зазвичай, на базі єдиного API.

Перевага при застосуванні фреймворків - вони мають стандартизовану структуру. З появою елементів інтерфейсу, які мали тенденцію до реалізації стандартної структури, фреймворки набули популярності. З їх використанням стало набагато простіше створювати засоби для автоматичного створення графічних інтерфейсів, оскільки структура внутрішньої реалізації коду програми стала відома заздалегідь. Для забезпечення каркасу, зазвичай, використовують підходи об'єктно-орієнтованого програмування, наприклад, частини програми можуть успадковуватися від базових класів фреймворка.

Плагін — додаток, незалежно скомпільований програмний модуль, що динамічно підключається до основної програми, призначений для розширення або використання її можливостей. Належить до загального програмного класу додатків. Плагіни, зазвичай, виконуються у вигляді динамічних бібліотек.

Основна програма надає сервіси, які плагін може використовувати. До них належить надана плагіну можливість зареєструвати себе в основному додатку, а також протокол обміну даними з іншими плагінами. Плагіни є залежними від сервісів, що надаються основним додатком, і переважно окремо не

використовуються. На противагу їм, основний додаток незалежно оперує плагінами, надаючи кінцевим користувачам можливість динамічно додавати й оновлювати плагіни без необхідності внесення змін в основний додаток.

Система управління базами даних

Система управління базами даних (СУБД) - це програмний пакет, призначений для визначення, маніпулювання, отримання та управління даними в базі даних. СУБД, як правило, маніпулює самими даними, форматом даних, іменами полів, структурою запису та структурою файлів. Він також визначає правила перевірки та маніпулювання цими даними.

СУБД позбавляє користувачів обрамлення програм для підтримки даних. Мови запитів четвертого покоління, такі як SQL, використовуються разом із пакетом СУБД для взаємодії з базою даних.

Деякі інші приклади СУБД включають:

- MySQL
- SQL Server
- Oracle
- dBASE
- FoxPro

Створення віртуальної частини веб-додатку було реалізовано за допомогою таких технологій: HTML, CSS, JavaScript, PHP. Серверна частина виконана на MySQL.

HTML (мова розмітки гіпертексту) - це код, який використовується для структури веб-сторінки та її вмісту. Наприклад, вміст може бути структурований у межах набору абзаців, списку пунктів з маркуванням або за допомогою зображень та таблиць даних.

Каскадні таблиці стилів (CSS) - мова сторінок стилів, яка використовується для опису подання документа, написаного мовою розмітки. Найчастіше використовується для встановлення візуального стилю веб-сторінок та інтерфейсів користувача, написаних у HTML та XHTML, мова може бути застосована до будь-

якого документа XML, включаючи звичайний XML, SVG та XUL, і застосовна для візуалізації в мовленні чи інших ЗМІ.

Приклад CSS коду:

```
body {  
  background-image: url(images/bg.gif);  
  background-repeat: no-repeat;  
  background-position: left top;  
}
```

JavaScript, JS, - мова програмування, яка відповідає специфікації ECMAScript.

JavaScript - це високорівнева та багатопарадигмальна мова програмування. Вона має синтаксис фігурної дужки, динамічне введення тексту, орієнтовану на прототип об'єктну орієнтацію та функції першого класу.

Поряд із HTML та CSS, JavaScript є однією з основних технологій всесвітньої павутини. JavaScript включає інтерактивні веб-сторінки і є невід'ємною частиною веб-додатків. Переважна більшість веб-сайтів використовують його для поведінки на стороні клієнта, а всі основні веб-браузери мають спеціальний механізм JavaScript для його виконання.

Приклад JavaScript коду:

```
function factorial(n) {  
  if (n === 0)  
    return 1; // 0! = 1  
  
  return n * factorial(n - 1);  
}  
factorial(3); // returns 6
```

PHP (рекурсивна аббревіатура для PHP: Hypertext Preprocessor) - широко використовувана сценарійна мова загального призначення з відкритим кодом, яка особливо підходить для веб-розробки та може бути вбудована в HTML.

Фреймворк WordPress - це онлайн-інструмент створення веб-сайтів з відкритим кодом, написаний на PHP. Особливістю фреймворку є те, що він включає архітектуру плагінів і систему шаблонів, що в WordPress називаються темами.

Щоб функціонувати, WordPress має бути встановлений на веб-сервері, або в частині веб-сервісу хостингу, як WordPress.com, або на комп'ютері, на якому працює програмний пакет WordPress.org, щоб самостійно функціонувати як хост мережі. Локальний комп'ютер може використовуватися для тестування та навчання для користувачів.

2.2. Вибір системи управління базою даних (СУБД)

Оскільки програмним продуктом є плагін, який працює в фреймворку WordPress, то використовується саме ця база даних.

У базі даних WordPress зберігаються всі необхідні дані веб-сайту. Не лише основна інформація, як імена користувачів та паролі, але повідомлення, сторінки та коментарі, навіть тема веб-сайту та налаштування WordPress.

База даних - це структурований набір організованих даних. В області ІТ системи управління базами даних (СУБД) посиляються на програмне забезпечення, яке взаємодіє з користувачем, даними та отримує та зберігає дані.

Існує кілька способів використання та доступу до бази даних. WordPress використовує MySQL для своєї системи управління базами даних.

MySQL - це програмне забезпечення з відкритим кодом, і воно відповідає за управління компонентами бази даних WordPress, таких як дані користувачів, мета користувача, повідомлення, коментарі тощо.

WordPress використовує PHP, використовуючи SQL-запит у межах розмітки PHP, як мову навчання даних CRUD (Створення, читання, оновлення та видалення) з бази даних MySQL. База даних є невід'ємною частиною WordPress. Саме магістраль зберігає всю основну інформацію.

3. ПРАКТИЧНА РЕАЛІЗАЦІЯ

3.1 Розробка моделі інформаційної системи

Інформаційна система (ІС) - це система, яка реалізує інформаційну модель предметної області, найчастіше будь-якої сфери людської діяльності. ІС повинна забезпечувати: отримання (введення або збір), зберігання, пошук, передачу та обробку даних, інформації.

Проектування моделі інформаційної системи є важливим етапом у розробці сучасного веб-додатку. Такий програмний продукт неможливий без процесу моделювання кожної складової, що складають систему.

При розробці інформаційної моделі було поставлено завдання розробити плагін для розміщення рекламних оголошень та впровадити його в фреймворк WordPress.

Метою інформаційної моделі програмного додатку є створення реклами для збільшення кількості рейтингу pr.

На рисунку 3.1 приведена схема роботи ІС, що розробляється.



Рис. 3.1 – Схема роботи ІС

По-перше, визначимо звідки надходить інформація. Інформацією в розумінні даної дипломної роботи виступає рекламна продукція, а саме оголошення. В сучасній сфері інформаційних технологій рекламні повідомлення набувають різних форм та форматів. Відео, фото, аудіо розширення зустрічаються не тільки в постах, тепер так виглядають оголошення у мережі Інтернет. Також повідомлення можна реалізовувати програмним чином засобами JavaScript та багатьма іншими способами. Підбиваючи підсумок з даного пункту, слід зазначити, що система, яка розробляється повинна забезпечити введення різних доступних на даний момент варіантів рекламних оголошень.

По-друге, зберігання інформації дуже вагомий аспект роботи. Змінюються часи, властивості, зовнішній вигляд рекламної компанії. Швидкоплинність часу впливає на хід подій, тому важливо забезпечити в ІС, що розробляється надійне зберігання даних та зручний пошук, що дозволить полегшити роботу адміна у користуванні плагіном.

Передача та обробка даних здійснюється засобами фреймворку WordPress. Саме на цій платформі розміщується плагін. WordPress надає своєрідний майданчик для розміщення реклами, у вигляді шаблонів сайтів. Платформа є популярною у користуванні не тільки програмістами, а й звичайними веб-користувачами.

Таким чином, раціональне розподілення функціоналу на етапі проектування є запорукою успіху в подальшій програмній реалізації. Адже, детальний аналіз можливостей та існуючих технологій створює добре спроектований продукт.

На рисунку 3.3 подана діаграма користувачів. Можна користувачів розділити на «Клієнт» та «Адмін».

На рисунку 3.2 зображена схема реалізації, згідно якої, клієнт має деякі свої права, адміністратор має всі права клієнта та доступ до панелі адміністратора, а сервер працює с базою даних та оброблює запити згідно дій користувача.



Рис. 3.2 - Діаграма користувачів

3.2 Програмна реалізація

Перед початком роботи необхідно встановити локальний сервіс для роботи з фреймворком. Для кращої роботи було встановлено XAMP. Після проведених маніпуляцій здійснюється перехід на офіційний сайт компанії WordPress. На данному Інтернет ресурсі є можливість завантажити архів з файлами, які по закінченню завантаження, необхідно перенести в папку локального сервісу «htdocs». Після активації локального сервісу, необхідно запустити WordPress. По завершенню реєстрації відбувається створення баз даних. Таким чином, майданчик для розміщення готовий.

Нижче наведені блоки програмного коду із файлу class – my.php.

Фрагмент, де відбувається отримання токена, url сторінки та підключення плагіну:

```

public function __construct ( $file ) {
    $this->token = 'asd';
    $this->plugin_url = trailingslashit( plugins_url( "", $plugin = $file ) );

    $this->load_plugin_textdomain();
    add_action( 'init', array( &$this, 'load_localisation' ), 0 );
  }

```

Визначення типу посту:

```
require_once( 'class-my-admin-settings.php' );
require_once( 'class-my-admin.php' );
$this->admin_inter = new ASD_Interst_Admin($file);
```

Оголошення типу рекламного оголошення:

```
add_action( 'wp_enqueue_scripts', array( &$this, 'enqueue_styles' ) );
add_action( 'wp_footer', array( &$this, 'enqueue_scripts' ) );
add_action( 'get_header', array( &$this, 'set_cookie' ) );
```

Завантаження плагіну:

```
public function load_localisation () {
    load_plugin_textdomain( 'my-ads', false, dirname( plugin_basename( $this->file ) ) . '/lang/' );
}

public function load_plugin_textdomain ()
{
    $domain = 'my-ads';
    // The "plugin_locale" filter is also used in load_plugin_textdomain()
    $locale = apply_filters( 'plugin_locale', get_locale(), $domain );
    load_textdomain( $domain, WP_LANG_DIR . '/' . $domain . '/' . $domain . '-' . $locale . '.mo' );
    load_plugin_textdomain( $domain, FALSE, dirname( plugin_basename( $this->file ) ) .
'/lang/' );
}
```

Підключення стилів:

```
public function enqueue_styles ()
{
    wp_register_style( $this->token . '-my', esc_url( $this->plugin_url . 'css/my.css' ), 'all' );
}
```

Підключення файлу .js.

```
public function enqueue_scripts ()
{
    wp_register_script( $this->token . '-count', esc_url( $this->plugin_url .
'/js/kkcountdown.min.js' ), array( 'jquery' ), '1.0', true );
    wp_register_script( $this->token . '-my', esc_url( $this->plugin_url . 'js/my.js' ), array(
'jquery' ), '1.0', true );}
```

Після підключення плагіну, потрібно визначити місце розміщення реклами на сторінці. Блоки коду демонструють варіанти підключення на домашній, додатковій сторінках.

```

class ASD_Interst_Functions
{
    protected function is_ads ( $id )
    {
        $args = array(
            'post_type'    => 'ads',
            'post_status'  => 'publish',
            'order'         => 'DESC',
            'orderby'      => 'date',
            'posts_per_page' => 1);

        $queryin = new WP_Query( $args );
        $cicle = true;

        while ( $queryin->have_posts() && $cicle ):
            $queryin->the_post();
            $valid = true;

            $is_cookie = ( isset( $_COOKIE["ads"] ) && !empty(
$_COOKIE["ads"] ) ) ? maybe_unserialize( stripslashes($_COOKIE["ads"]) ) : "";
            $want = get_post_meta( get_the_ID(), 'post_once', true );

            if( $valid && ( !empty( $is_cookie ) && in_array( get_the_ID(),
$is_cookie ) && !empty( $want ) && $want == 'yes' ) ){
                $valid = false;
            }

            $is_as_html = get_post_meta( get_the_ID(), 'include_html', true
);

            if( $valid && empty( $is_as_html ) ){
                $valid = false;}
    }
}

```

Вибір розташування рекламного поста.

```

    $where = get_post_meta( get_the_ID(), 'where_show', true );

    if( $valid && $where == 'show_home' )
    {
        if( !is_home() && !is_front_page() )
        {
            $valid = false;
        }
    }

    if( $valid && $where == 'show_cust' && !is_home() &&
!is_front_page() ){
        $ads_posts = get_post_meta( get_the_ID(), 'ads_posts', true );
        if(!empty($ads_posts))
        {
            $sids = explode( ",", $ads_posts );
            if( !in_array( $id, $sids ) )
            {
                $valid = false;
            }
        }
    }

```

Приклад розміщення на додатковій сторінці, не включаючи, домашню:

```

if( $valid && $where == 'show_cust' && ( is_home() || is_front_page() ) )
{
    $valid = false;
}

if( $valid )

```

```

{
    $sicle = false;
    return get_the_ID();
}

endwhile;
wp_reset_query();
}

```

Після проведених операцій, необхідно зайти в вкладку плагіни в WordPress і здійснити активацію плагіну “My Ads”.

Для того, щоб додати нове рекламне оголошення необхідно натиснути “Add New Ad”, в результаті відобразиться наступне:

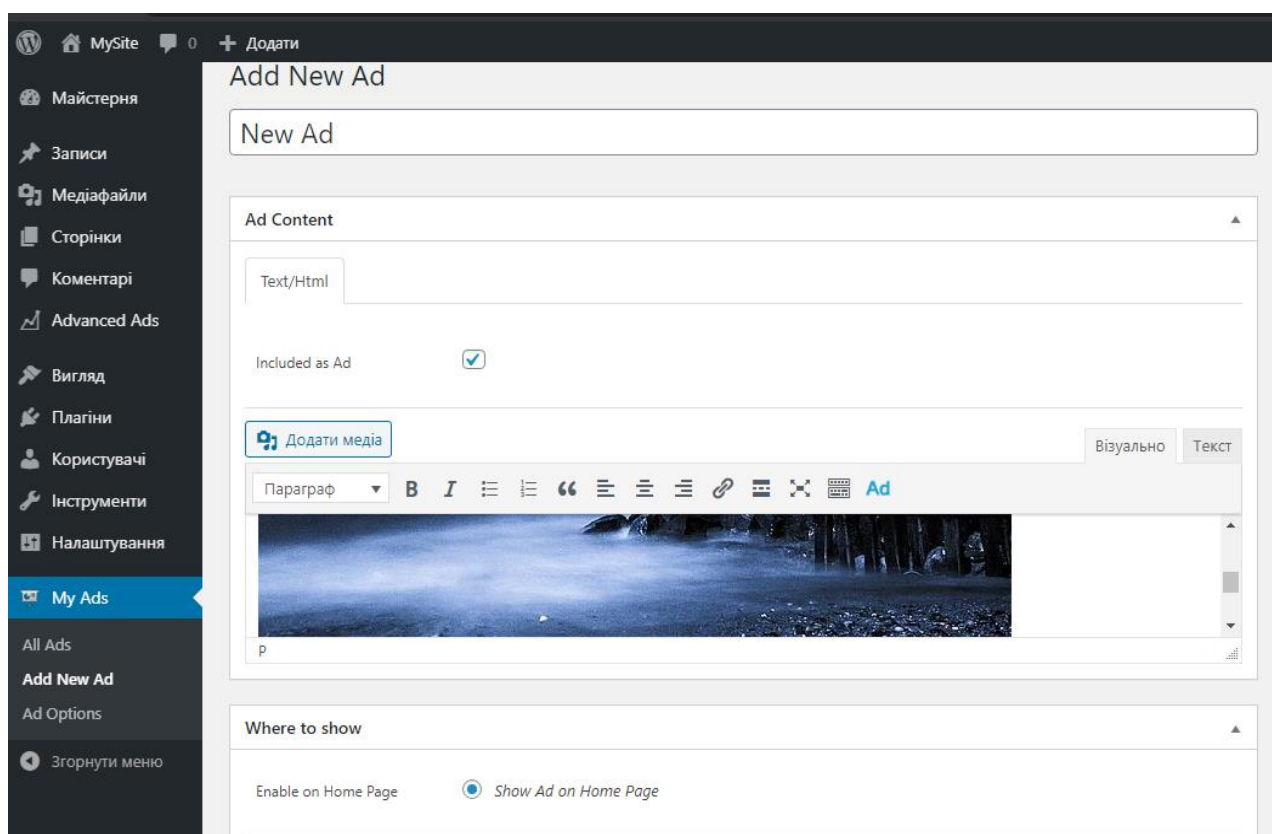


Рис 3.3 – Вкладка додавання рекламного оголошення

Для зміни параметрів відображення рекламного посту, необхідно активувати вкладку “Ad Options”

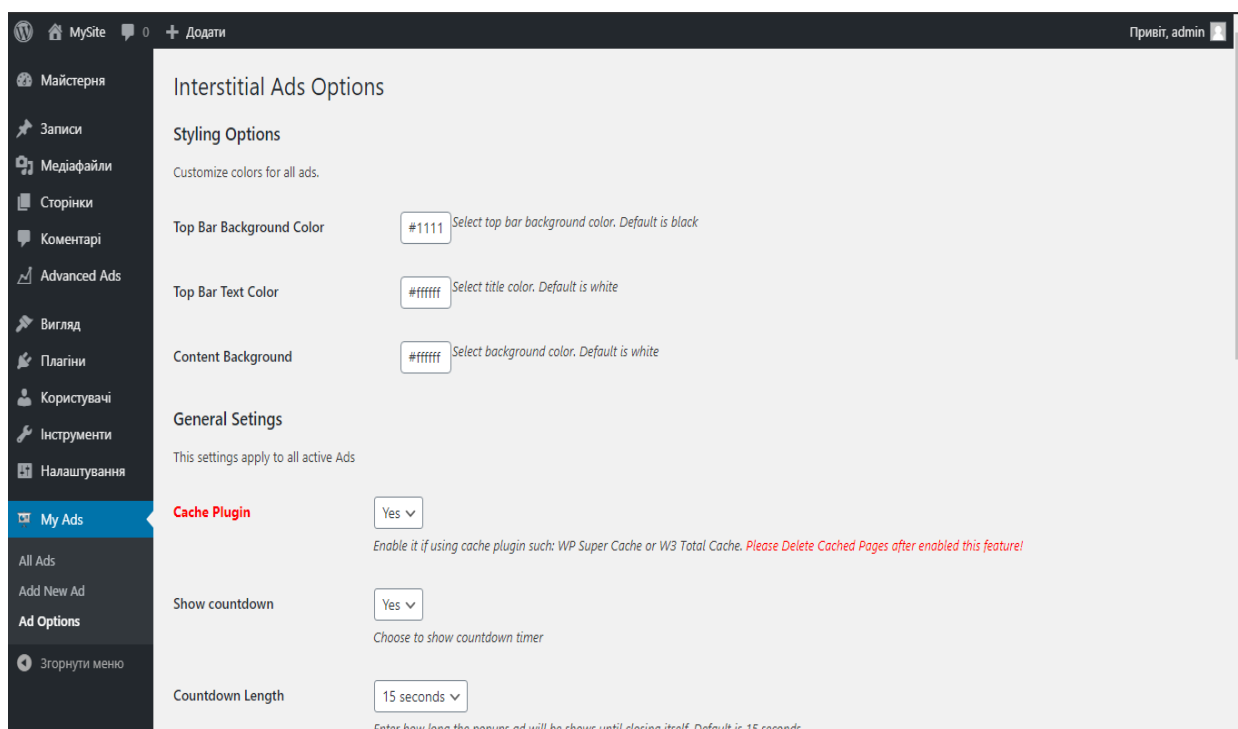


Рис. 3.4 -Вкладка “Ad Options”

Плагін також надає такі можливості, як перегляд вже створеної реклами, її редагування та видалення:

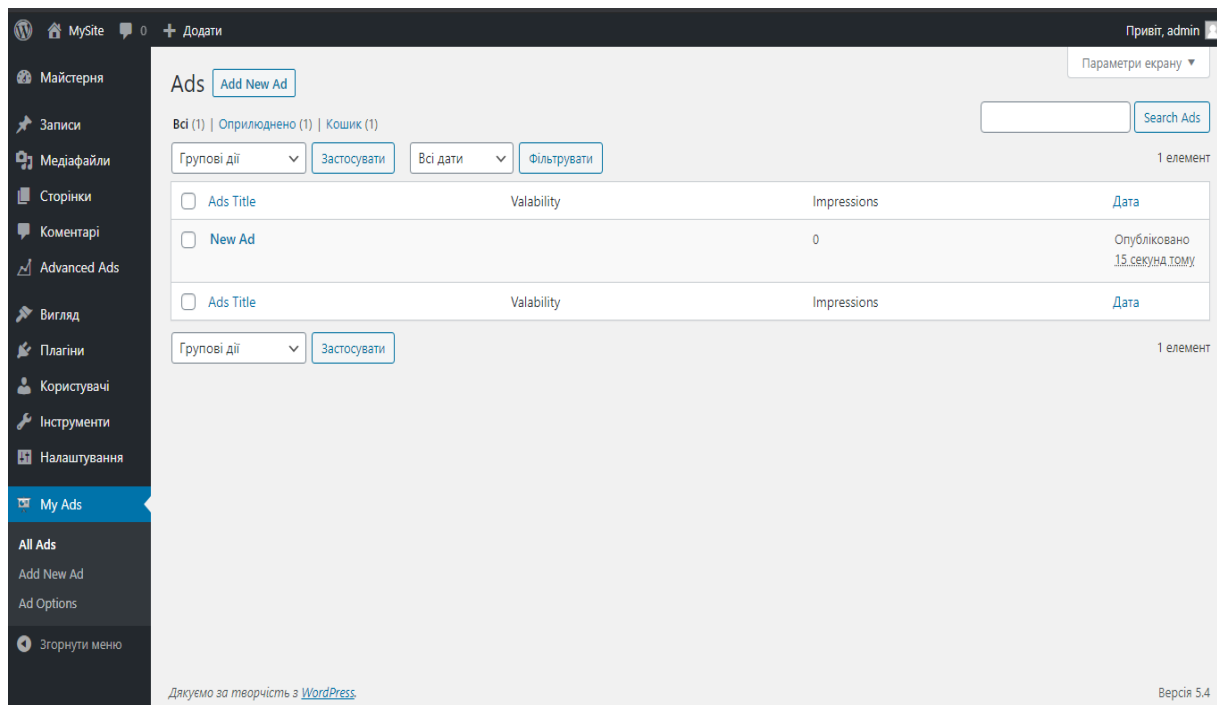


Рис. 3.5 - Вкладка “All Ads”

Так виглядає реклама у вигляді відео. Після закінчення часу її можна закрити. Також варто зауважити, що таку рекламу можна вставити як із медіабібліотеки користувача, так і посиланням.

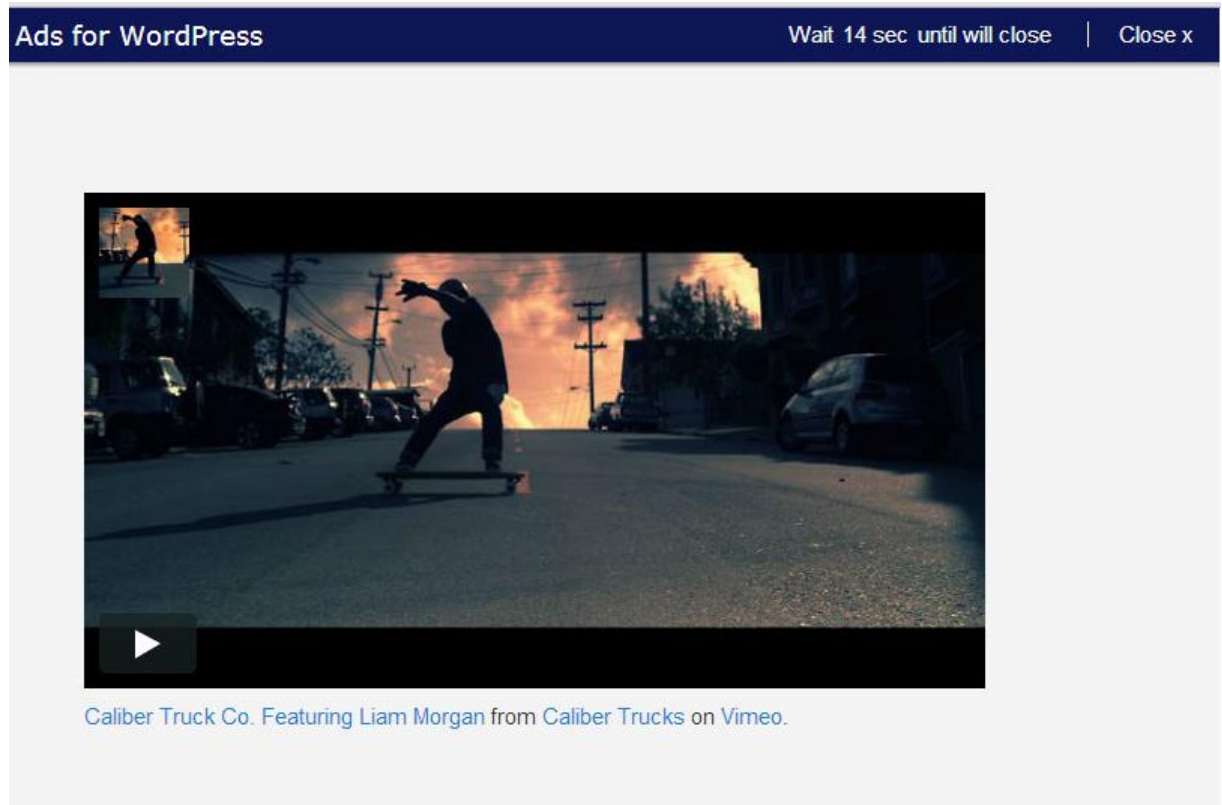


Рис. 3.6 – Відео-реклама

3.3 Безпека web-ресурсу

Сьогоднішні реалії створили нові умови для розповсюдження інформації в Інтернеті. Тепер потрібно не тільки обачливо ставитися до підозрілих ресурсів, але й захищати свої персональні дані від дій зловмисників.

Реклама є гарним методом для просування товару чи послуги засобами всесвітньої мережі. Проте вигідні пропозиції іноді становлять загрозу користувачам у вигляді матеріальних і моральних збитків. Так посилання сумнівного характеру може призвести втрату фінансового характеру шляхом викрадення банківських даних.

При проектуванні та створенні плагіну була поставлена мета – розробка не тільки якісного майданчику, але й максимально безпечного у використанні. Описання засобів, які були використані наведено нижче.

Оскільки, плагін реалізовано для фреймворку WordPress, щоб підвищити індекс цитування сайту та прорекламувати товар, то накладаються певні вимоги задля покращення безпеки.

По-перше, при реєстрації нового користувача необхідно використовувати унікальний логін та складний пароль (не менше 8ми неповторюваних символів). А також один із важливих аспектів цього пункту – користувача admin потрібно видалити із списку. Також в налаштуваннях необхідно обмежити кількість невдалих спроб авторизації. Ці превентивні заходи допоможуть уникнути атак методом підбору.

По-друге, варто зазначити, що майданчик для розміщення реклами, а саме фреймворк повинен оновлюватися. Так із регулярними оновлення безпека даних та репутація рекламодавця не постраждає. Один важливий нюанс для користувача при майбутньому користуванні – важливо приховувати поточну версію фреймворку. Для уникнення хакерського втручання.

По-третє, створений .htaccess файл, де зібрані необхідні варіації кодів, що блокують доступ до wp-config.php файлу, що містить важливу інформацію для з'єднання до серверу MySQL та бази даних.

```
<Files wp-config.php>
order allow, deny
deny from all
</Files>
```

Також наявний код, що захищає від SQL-ін'єкцій. Одна із найактуальніших видів атак на сайти WordPress.

```
RewriteCond %{QUERY_STRING} (<|%3C).*script.*(>|%3E) [NC, OR]
RewriteCond %{QUERY_STRING} GLOBALS(=|\\|%0 - 9A-Z){0, 2}) [OR]
RewriteCond %{QUERY_STRING} _REQUEST(=|\\|%0 - 9A-Z) {0,2})
RewriteRule ^(.*)$ index.php [F,L]
```

Захист від хотлінкінга.

```
RewriteEngine On
RewriteCond %{HTTP_REFERER} !^http://(.\+\.)?yourdomain\.com/ [NC]
```

```
RewriteCond          %{HTTP_REFERER}             !^$  
RewriteRule .*\.(\.jpe?g|gif|bmp|png)$ /images/nohotlink.jpg [L]
```

Досить популярне явище не тільки для реклами, а й для контенту в цілому. Суть якого полягає в вставці зображення на сторонній ресурс, при цьому трафік має перехід на коректний сервер. За допомогою приведеного коду сервер здійснює перевірку звідки прямує зображення, якщо шлях вірний, користувач бачить зображення коректно. У зворотньому випадку – відображення зображення з дефектом. Такі заходи убезпечать від навмисного плагіату в шахрайських цілях.

Код, що обмежить доступ до файлу .htaccess.

```
<files .htaccess>  
order allow, deny  
deny from all  
</files>
```

Наведено приклад коду, який забороняє відслідковування HTTP заголовків:

```
RewriteEngine                On  
RewriteCond                  %{REQUEST_METHOD}   ^TRACE  
RewriteRule .* – [F]
```

Код, який захищає від використання XSS-ін'єкцій і спроби модифікувати змінні GLOBALS і _REQUEST:

```
Options                        +FollowSymLinks  
RewriteEngine                On  
RewriteCond                  %{QUERY_STRING}    (<|<%3C).*script.*(>|%3E)    [NC,OR]  
RewriteCond                  %{QUERY_STRING}    GLOBALS(=\\[\\%[0-9A-Z]{0,2})    [OR]  
RewriteCond                  %{QUERY_STRING}    _REQUEST(=\\[\\%[0-9A-Z]{0,2})  
RewriteRule ^(.*)$ index.php [F,L]
```

Фактору якому також слід приділяти увагу протоколу при з'єднанні з ресурсом. Введеному випадку використовується протокол HTTPS, який забезпечує автентифікацію та шифрування комунікацій. Такий підхід є надважливим у платіжних системах та корпоративних логінах.

Підбиваючи підсумок, варто зазначити, що приведені методи захисту ресурсу є необхідними в сьогоднішніх реаліях користування послугами мережі Інтернет.

ВИСНОВКИ

Таким чином, у ході дипломної роботи було створено проект, що являє собою рекламу в мережі Інтернет, яка реалізована у вигляді плагіну для фреймворку WordPress. Ця платформа є досить актуальним інструментом у створенні сайтів. На ній написані такі відомі сайти, як Adobe, Flickr, Tribune Media Group. Тому рекламні пости швидко знайдуть свого потенціального клієнта.

Рекламне оголошення є необхідним для поширення інформації та донесення її сутності до користувача. Можна рекламувати не тільки товари, але й певні ідеї. Так, існує алгоритм просування в глобальній мережі Інтернет, що є необхідним для підвищення рейтингу та індексу цитування сайту білим методом. Що дозволяє швидко, а головне якісно оптимізувати web-ресурс без впливу на пошукові алгоритми, що блокуються.

Плагін створює належні умови для розробки власного рекламного оголошення. Наявна фільтрація, вибір сторінок, налаштування зрозумілі для користувача із будь-яким рівнем користування ПК.

Належна увага приділена безпеці користування та збереження персональних відомостей користувачів. Використання протоколу HTTPS забезпечить надійне шифрування даних при її передачі. Впроваджена низка програмних кодів, які захищають від кібер атак та блокують шахрайську діяльність, впроваджену сторонніми ресурсами.

Інтернет простір став буденністю для сьогодення. Використання мережі не являє собою надзвичайного явища, реклама набула персоналізації і отримала свій відсоток довіри. Проте, не варто нехтувати статистикою кібер шахрайств, що призводять до комерційних втрат. Даний рекламний плагін для фреймворку WordPress – безпечний алгоритм просування товару чи сайту, що створить сприятливі умови для користування.

Необхідно зазначити, що реалізація даного рекламного продукту неможлива без підключення до мережі Інтернет та фреймворку WordPress.

СПИСОК ЛІТЕРАТУРИ

1. Янсен, Б. Дж., Маллен, Т. Спонсорський пошук: огляд концепції, історії та технології: міжнародний журнал електронного бізнесу, 2008, 114–131 с.
2. Kaifu Z. Контекстуальна реклама / Z. Kaifu, K. Zsolt. – 2012. – №6. – с. 980–994.
3. Г. Ю. Філімонов, С. А. Цатурян. Соціальні мережі як інноваційний механізм «м'якого» впливу і управління масовою свідомістю // «Політика і суспільство». - М .: Nota Bene, 2012. - С. 65 - 75.
4. Ігор Ашманов, Андрій Іванов. Оптимізація і просування сайтів в пошукових системах. — СПб. : Пітер, 2009. — с. 400.
5. Системи електронної контент-комерції : монографія / А. Ю. Берко, В. А. Висоцька, В. В. Пасічник ; М-во освіти і науки України, Нац. ун-т «Львів. політехніка». – Л. : Вид-во Нац. ун-ту «Львів. політехніка», 2009. – 610 с.
6. Персоналізовані та неперсоналізовані оголошення [Електронний ресурс] – Режим доступу до ресурсу: - <https://support.google.com/adsense/answer/9007336?hl=uk>.
7. Google Ads [Електронний ресурс] – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Google_Ads.
8. Яндекс. Директ [Електронний ресурс] – Режим доступу до ресурсу: <https://direct.yandex.com.ua>.
9. WordPress [Електронний ресурс] – Режим доступу до ресурсу: <https://wordpress.org>.
10. Власов К. А. Total Commander: ефективна робота з файлами і архівами. - БХВ-Петербург, 2012. - Глава 24. Модулі. - 752 с.
11. Мова програмування JavaScript [Електронний ресурс] – Режим доступу до ресурсу: <https://learn.javascript.ru/js>.
12. Мова програмування PHP [Електронний ресурс] – Режим доступу до ресурсу: <https://www.php.net/manual/en/language.types.type-juggling.php>.
13. MySQL [Електронний ресурс] – Режим доступу до ресурсу: <http://www.mysql.com>.

14. HTML [Электронный ресурс] – Режим доступа до ресурсу:
<https://www.w3.org/TR/html401/>.
15. CSS [Электронный ресурс] – Режим доступа до ресурсу:
<https://www.w3schools.com/css/default.asp>.
16. JavaScript [Электронный ресурс] – Режим доступа до ресурсу:
<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference>.

ДОДАТКИ

ДОДАТОК А

Лістинг коду файла class – my.php.

```

<?php
if ( ! defined( 'ABSPATH' ) ) exit;

class ASD_Interest extends ASD_Interst_Functions{
    var $version;
    private $file;
    public $keep;

    public function __construct ( $file ) {
        $this->token = 'asd';
        $this->plugin_url = trailingslashit( plugins_url( "", $plugin = $file ) );

        $this->load_plugin_textdomain();
        add_action( 'init', array( &$this, 'load_localisation' ), 0 );

        require_once( 'class-my-admin-settings.php' );
        require_once( 'class-my-admin.php' );
        $this->admin_inter = new ASD_Interst_Admin($file);

        add_action( 'wp_enqueue_scripts', array( &$this, 'enqueue_styles' ) );
        add_action( 'wp_footer', array( &$this, 'enqueue_scripts' ) );

        add_action( 'get_header', array( &$this, 'set_cookie' ) );

        if( $this->is_cached( ) ){
            add_action( 'wp_ajax_my_ads_action', array( &$this, 'my_ajax_ads' ) );
            add_action( 'wp_ajax_nopriv_my_ads_action', array( &$this, 'my_ajax_ads' ) );
        }

        add_action( 'wp_footer', array( &$this, 'my_ads' ) );
    }
}

```

```

public function load_localisation () {
    load_plugin_textdomain( 'my-ads', false, dirname( plugin_basename( $this->file ) ) . '/lang/' );
}
public function load_plugin_textdomain () {
    $domain = 'my-ads';

    $locale = apply_filters( 'plugin_locale', get_locale(), $domain );
    load_textdomain( $domain, WP_LANG_DIR . '/' . $domain . '/' . $domain . '-' . $locale . '.mo' );
    load_plugin_textdomain( $domain, FALSE, dirname( plugin_basename( $this->file ) ) . '/lang/' );
}
public function enqueue_styles () {
    wp_register_style( $this->token . '-my', esc_url( $this->plugin_url . 'css/my.css' ), 'all' );
}
public function enqueue_scripts () {
    wp_register_script( $this->token . '-count', esc_url( $this->plugin_url . 'js/kkcountdown.min.js' ), array( 'jquery' ), '1.0', true );
    wp_register_script( $this->token . '-my', esc_url( $this->plugin_url . 'js/my.js' ), array( 'jquery' ), '1.0', true );
}
public function set_cookie () {
    if ( !is_admin() ){
        global $post;
        $ad_id = $this->is_myads( $post->ID );

        if( $ad_id )
            $this->let_cookie( $ad_id );
    }
}
public function my_ads ($id = NULL) {
    if($id){
        $id_post = $id;
    }
}

```



```

} else {
    global $post;
    $id_post = $post->ID;
}
$ad_id = $this->is_myads( $id_post );

$settings = array(
    'bar_color'    => 'bar_color',
    'title_color' => 'title_color',
    'back_color'  => 'back_color',
    'cont_width'  => 'cont_width',
    'close_name'  => 'but_close',
    'show_timer'  => 'show_timer',
    'countdown_time' => 'countdown_time',
    'wait_time'   => 'wait_time',
    'tra_close'   => 'tra_close',
    'tra_wait'    => 'tra_wait',
    'tra_seconds' => 'tra_seconds',
    'tra_minutes' => 'tra_minutes',
    'tra_until'   => 'tra_until'
);

foreach($settings as $set => $name){
    ${$name} = $this->int_option( $set );
}

if( $this->is_cached( ) ){

    $JSmyAds = array( 'is_cached' => 'true', 'id_post' => $id_post, 'ajaxurl' => admin_url( 'admin-
ajax.php' ) );

    if ( !empty( $wait_time ) && ( int ) $wait_time > 0 ){
        $JSmyAds = array_merge( $JSmyAds, array( 'is_wait' => $wait_time ) );
    }
}

```

```

if ( !empty( $countdown_time ) && ( int ) $countdown_time > 0 && $show_timer == 'yes' ){
    wp_enqueue_script( $this->token . '-count' );
    $JSmyAds = array_merge($JSmyAds, array( 'is_count' => $countdown_time ));
    $add_wait = ( !empty( $wait_time ) && ( int ) $wait_time > 0 ) ? ( int ) $wait_time : 0;
    $countdown = time() + $add_wait + ( int ) $countdown_time;

}

if ( !empty( $tra_seconds ) && $show_timer == 'yes' ){
    $JSmyAds = array_merge( $JSmyAds, array( 'seconds' => $tra_seconds ));
}

if ( !empty( $tra_minutes ) && $show_timer == 'yes' ){
    $JSmyAds = array_merge( $JSmyAds, array( 'minutes' => $tra_minutes ));
}

wp_enqueue_style( $this->token . '-my' );
wp_enqueue_script ( $this->token . '-my' );
wp_localize_script( $this->token . '-my', 'myAds', $JSmyAds );

}

if( $ad_id ){
    $this->impression( $id_post );

    $html_ad = $style_main = $style_cont = "";

    $cont_width = !empty( $cont_width ) ? 'width:'. $cont_width .'px;' : "";
    $back_color = !empty( $back_color ) ? 'background:'. $back_color .';' : "";
    $bar_color = !empty( $bar_color ) ? 'background:'. $bar_color .';' : "";
    $tit_color = !empty( $title_color ) ? 'color:'. $title_color .';' : "";
    $is_wait = !empty( $wait_time ) && ( int ) $wait_time > 0 ? 'display: none;' : "";

    $col_style = !empty( $tit_color ) ? 'style="'. $tit_color .'" : ";
    $bor_style = !empty( $title_color ) ? 'style="border-color:'. $title_color .';'" : ";

```

```

$bor_style = ( !empty( $but_close ) && $but_close == 'no' ) ? 'style="border:none;" : ";

if( !empty( $back_color ) || !empty( $is_wait ) ){
    $style_main = 'style="'. $back_color . $is_wait ."'";
}

if( !empty( $cont_width ) ){
    $style_cont = 'style="'. $cont_width ."'";
}

if( !empty( $bar_color ) || !empty( $tit_color ) ){
    $style_bar = 'style="'. $bar_color . $tit_color ."'";
}

if( !$this->is_cached() ){
    $JSmyAds = array();
    if ( !empty( $wait_time ) && ( int ) $wait_time > 0 ){
        $JSmyAds = array( 'is_wait' => $wait_time );
    }
    if ( !empty( $countdown_time ) && ( int ) $countdown_time > 0 && $show_timer == 'yes' )
{
        wp_enqueue_script( $this->token . '-count' );
        $JSmyAds = array_merge($JSmyAds, array( 'is_count' => $countdown_time ));
        $add_wait = ( !empty( $wait_time ) && ( int ) $wait_time > 0 ) ? ( int ) $wait_time : 0;
        $countdown = time() + $add_wait + ( int ) $countdown_time;

    }

    if ( !empty( $tra_seconds ) && $show_timer == 'yes' ){
        $JSmyAds = array_merge( $JSmyAds, array( 'seconds' => $tra_seconds ));
    }

    if ( !empty( $tra_minutes ) && $show_timer == 'yes' ){
        $JSmyAds = array_merge( $JSmyAds, array( 'minutes' => $tra_minutes ));
    }

    if ( !$this->is_cached() ){

```

```

        $JSmyAds = array_merge( $JSmyAds, array( 'id_post' => $id_post, 'ajaxurl' => admin_url
( 'admin-ajax.php' ) ) );
    }

    if ( !empty( $JSmyAds ) ){
        wp_localize_script( $this->token . '-mymy', $JSmyAds );
    }

    wp_enqueue_style( $this->token . '-my' );
    wp_enqueue_script( $this->token . '-my' );
}

$html_ad .= '<!-- @wpmanage.com-->
    <div id="my" '. $style_main .'>
    <div id="my-bar" '. $style_bar .'>
        <div id="my-tit" class="my">'. $this->get_myad( $ad_id, 'title' ) .'</div>
        <div id="my-close">';

if( !empty( $show_timer ) && $show_timer == 'yes' )
    $html_ad .= '<div id="my-mess" '. $bor_style .'>
        <span> '. $tra_wait .' </span>
        <span time="'. $countdown .'" class="my-kkcount-down"></span>
        <span> '. $tra_until .' </span>
    </div>';

if( !empty( $but_close ) && $but_close == 'yes' )
    $html_ad .= '<a href="javascript:return false" onclick="myads_close();" '. $col_style .'> '. $tra
_close .' <span> x </span></a>';
$html_ad .= '</div>
</div>
<div id="my-cnt" '. $style_cont .'>
    '. $this->get_myad( $ad_id ) .'
</div>
</div>';

```

```
if( !empty( $html_ad ) && !$this->is_cached() ){  
    echo $html_ad;  
} else if( !empty( $html_ad ) && $this->is_cached() ) {  
    return $html_ad;  
}  
  
}  
  
}  
  
}  
?>
```

ДОДАТОК Б

Лістинг коду файла Class-my-admin.php

```
<?php
if ( ! defined( 'ABSPATH' ) ) exit;

class Uji_Interst_Admin extends Uji_Interst_Admin_API{
    var $version;
    private $file;
    public static $plugin_url;
    public static $plugin_path;

    public function __construct ( $file ) {
        parent::__construct();

        $this->token = 'asd';
        $this->page_slug = 'asd-api';
        $this->opt_name = __( 'My Ads Options', 'asd' );

        $this->post_meta = 'asd_meta';

        self::$plugin_url = trailingslashit( plugins_url( "", $plugin = $file ) );
        self::$plugin_path = trailingslashit( dirname( $file ) );

        $this->labels = array();
        $this->setup_post_type_labels_base();

        add_action( 'init', array( &$this, 'add_post_type_ads' ), 100 );

        add_filter( 'manage_edit-asd_columns', array( &$this, 'add_column_headings' ), 10, 1 );
        add_action( 'manage_posts_custom_column', array( &$this, 'add_column_data' ), 10, 2 );

        add_filter( 'enter_title_here', array( &$this, 'change_default_title' ) );
```

```

add_action( 'admin_enqueue_scripts', array( &$this, 'admin_styles_ads' ) );
add_action( 'admin_print_scripts', array( &$this, 'admin_enqueue_scripts' ) );
add_action( 'admin_menu', array( &$this, 'asd_menu' ) );

```

```

add_action( 'add_meta_boxes', array( &$this, 'asd_meta_boxes' ) );
add_action( 'save_post', array( &$this, 'asd_save' ) );

```

```

add_action( 'admin_init', array( &$this, 'remove_add' ) );

```

```

}

```

```

private function setup_post_type_labels_base () {

```

```

    $this->labels = array( 'asd' => array() );

```

```

    $this-

```

```

>labels['asd'] = array( 'singular' => __( 'Ad', 'asd' ), 'plural' => __( 'Ads', 'asd' ), 'menu' => __( 'My Ads'
, 'asd' ) );

```

```

}

```

```

public function add_post_type_ads () {

```

```

    $args = array(

```

```

        'labels' => $this->create_post_type_labels( 'asd', $this->labels['asd']['singular'], $this-
>labels['asd']['plural'], $this->labels['asd']['menu'] ),

```

```

        'public' => false,

```

```

        'publicly_queryable' => true,

```

```

        'show_ui' => true,

```

```

        'show_in_menu' => true,

```

```

        'query_var' => true,

```

```

        'rewrite' => array( 'slug' => 'asd', 'with_front' => false, 'feeds' => false, 'pages' => false ),

```

```

        'capability_type' => 'post',

```

```

        'has_archive' => false,

```

```

        'hierarchical' => false,

```

```

        'menu_position' => 100,

```

```

        'supports' => array( 'title' )

```

```

);

register_post_type( 'asd', $args );
}
public function add_column_headings ( $defaults ) {
    $new_columns['cb'] = '<input type="checkbox" />';
    $new_columns['title'] = _x( 'Ads Title', 'column name', 'asd' );
    $new_columns['valability'] = _x( 'Valability', 'column name', 'asd' );
    $new_columns['impress'] = _x( 'Impressions', 'column name', 'asd' );

    if ( isset( $defaults['date'] ) ) {
        $new_columns['date'] = $defaults['date'];
    }

    return $new_columns;
}
public function add_column_data ( $column_name, $id ) {
    global $wpdb, $post;

    switch ( $column_name ) {
        case 'id':
            echo $id;
            break;

        case 'valability':
            $value = __( 'Unlimited', 'asd' );
            $val_from = get_post_meta( $post->ID, '_datapick1', true );
            $val_to = get_post_meta( $post->ID, '_datapick2', true );

            if ( $val_from || $val_to ) {
                echo ( $val_from ) ? $val_from : ";
                echo ( $val_to ) ? (( $val_from ) ? ' - ' . $val_to : $val_to ) : ";
            }else{
                echo $value;
            }
        }
    }
}

```



```

    }
    break;

    case 'impress':
        $num = get_post_meta( $id, 'ads_impressions', true );
        echo ( !empty($num) ) ? $num : 0;
    break;

    default:
    break;
}
}

private function create_post_type_labels ( $token, $singular, $plural, $menu ) {
    $labels = array(
        'name' => sprintf( _x( '%s', 'post type general name', 'asd' ), $plural ),
        'singular_name' => sprintf( _x( '%s', 'post type singular name', 'asd' ), $singular ),
        'add_new' => sprintf( _x( 'Add New %s', $token, 'asd' ), $singular ),
        'add_new_item' => sprintf( __( 'Add New %s', 'asd' ), $singular ),
        'edit_item' => sprintf( __( 'Edit %s', 'asd' ), $singular ),
        'new_item' => sprintf( __( 'New %s', 'asd' ), $singular ),
        'all_items' => sprintf( __( 'All %s', 'asd' ), $plural ),
        'view_item' => sprintf( __( 'View %s', 'asd' ), $singular ),
        'search_items' => sprintf( __( 'Search %s', 'asd' ), $plural ),
        'not_found' => sprintf( __( 'No %s found', 'asd' ), strtolower( $plural ) ),
        'not_found_in_trash' => sprintf( __( 'No %s found in Trash', 'asd' ), strtolower( $plural ) ),
        'parent_item_colon' => "",
        'menu_name' => $menu
    );

    return $labels;
}

public function admin_styles_ () {
    $screen = get_current_screen();

```

```

if (in_array( $screen->id, array( 'asd', 'ads_page_ujiinter-api' ))) :

    wp_register_style( 'admin-asd', self::$plugin_url . 'css/admin.css', "", '1.0', 'screen' );
    wp_register_style( 'bootstrap', self::$plugin_url . 'assets/bootstrap/css/bootstrap.css', "", '2.0', 'screen' );
    wp_register_style( 'colorpicker', self::$plugin_url . 'assets/colorpicker/css/colorpicker.css', "", '1.0', 'screen' );
    wp_register_style( 'datapicker', self::$plugin_url . 'assets/datepicker/css/datepicker.css', "", '1.0', 'screen' );
    wp_enqueue_style( 'admin-asd' );
    wp_enqueue_style( 'bootstrap' );
    wp_enqueue_style( 'colorpicker' );
    wp_enqueue_style( 'datapicker' );

endif;
}
public function admin_enqueue_scripts () {

    $screen = get_current_screen();

    if (in_array( $screen->id, array( 'asd', 'ads_page_ujiinter-api' ))) :

        wp_enqueue_script( 'bootstrap', self::$plugin_url . 'assets/bootstrap/js/bootstrap.min.js', array( 'jquery' ), '2.0' );
        wp_enqueue_script( 'bootstrap-color', self::$plugin_url . 'assets/colorpicker/js/bootstrap-colorpicker.js', array( 'jquery' ), '1.0' );
        wp_enqueue_script( 'asd', self::$plugin_url . 'js/admin-asd.js', array( 'jquery' ), '1.0' );

    endif;

}
public function change_default_title( $title ){
    $screen = get_current_screen();

```

```

if ( 'asd' == $screen->post_type ) {
    $title = 'Enter Ad Title Here';
}

return $title;
}

public function remove_add( ){
    $published_posts = wp_count_posts( 'asd' );

    if( (int) $published_posts->publish > 0 ){
        remove_submenu_page( 'edit.php?post_type=asd', 'post-new.php?post_type=asd' );
        add_action('admin_footer', array( &$this, 'add_footer_css' ) );
    }

}

public function add_footer_css( ){
    $css = "";
    if ( isset($_GET['post_type']) && !empty($_GET['post_type']) && $_GET['post_type'] == 'asd' )
    {
        $css = '#favorite-actions {display:none;}
            .add-new-h2{display:none;}
            .tablenav{display:none;}';
    }
    echo '<style type="text/css">
        #wp-admin-bar-new-asd {display:none;}
        '. $css .'
    </style>';
}

public function ujinter_menu() {
    $hook = add_submenu_page( 'edit.php?post_type=asd', __( 'My Ads', 'asd' ), __( 'Ad Options', 'asd' ),
    'manage_options', $this->page_slug, array( &$this, 'settings_screen' ) );
    if ( isset( $_GET['page'] ) && ( $_GET['page'] == $this->page_slug ) ) {

```

```

        add_action( 'admin_notices', array( &$this, 'settings_errors' ) );

    }

}

public function asd_meta_boxes() {
    global $post;

    if ( function_exists('wp_editor') ) {
        remove_meta_box( 'postexcerpt', 'product', 'normal' );
        add_meta_box( 'postexcerpt', __('Ads Content', 'asd'), array( &$this, 'asd_html' ), 'asd', 'normal'
    );
    }

    add_meta_box( 'postwhere', __('Where to show', 'asd'), array( &$this, 'asd_where' ), 'asd', 'normal'
);
    add_meta_box( 'getpro', __('My Ads Premium', 'asd'), array( &$this, 'asd_prover' ), 'asd', 'side' );

}

public function asd_html( $post ) {
    ?>
    <ul class="nav nav-tabs" id="cont_tab">
        <li><a href="#int-tab-1" data-toggle="tab"><?php _e("Text/Html", 'asd') ?></a></li>
        <li><a href="#int-tab-4" data-toggle="tab"><?php _e("Settings", 'asd') ?></a></li>
    </ul>
    <div class="tab-content">
    <?php
    $include = get_post_meta( $post->ID, 'include_html', true );
    ?>
    <div class="tab-pane" id="int-tab-1">
        <div class="options_group" style="margin-bottom:8px;">

```

```

    <p class="form-field">
        <label for="include_html"><?php _e("Included as Ad", 'asd') ?></label>
        <input id="include_html" class="checkbox" type="checkbox" value="yes" name="includ
e_html" <?php checked( $include, 'yes' ) ?>>
    </p>
</div>
<?php
$settings = array(
    'quicktags' => array( 'buttons' => 'em,strong,link' ),
    'textarea_name' => 'excerpt',
    'quicktags' => true,
    'tinymce' => true,
    'editor_css' => '<style>#wp-excerpt-editor-container .wp-editor-
area{height:275px; width:100%;}</style>'
);

wp_editor( htmlspecialchars_decode( $post->post_excerpt ), 'excerpt', $settings );

echo '</div>';

$include = get_post_meta( $post->ID, 'include_url', true );
?>
<div class="tab-pane" id="int-tab-2">
    <div class="options_group">
        <p class="form-field">
            <label for="include_url"><?php _e("Included as Ads", 'asd') ?></label>
            <input id="include_url" class="checkbox" type="checkbox" value="yes" name="include_
url" <?php checked( $include, 'yes' ) ?>>
        </p>
    </div>
    <div class="options_group">
<?php
for($x=1; $x<=5; $x++){
    $include = get_post_meta( $post->ID, 'ads_link'.$x, true );

```

```

echo'<p class="form-field">
    <label for="ads_link">'. __("Add Link URL", 'asd') .$.x.'</label>
    <input type="text" name="ads_link'.$.x.'" class="medium" id="ads_link'.$.x.'" value="'.$.in
clude.'" />
    </p>;

}

echo '</div>
</div>';
?>
<div class="tab-pane" id="int-tab-3">
    <div class="options_group">
        <p class="form-field">
            <label for="ads_link"><?php _e("Selected Posts/Pages", 'asd') ?></label>
            <input type="text" name="add_posts" class="short" id="add_posts" value="<?php echo
get_post_meta( $post->ID, 'add_posts', true ); ?>" />
            <span class="description"><?php _e("Add any pages or posts id separated by commas.
ex: 312, 16, 27", 'asd') ?></span>
        </p>
    </div>
</div>
<?php

$.include = get_post_meta( $post->ID, 'show_title', true );
$.include_one = get_post_meta( $post->ID, 'post_random', true );
$.include_two = get_post_meta( $post->ID, 'post_once', true );
?>
<div class="tab-pane" id="int-tab-4">
    <div class="options_group">
        <p class="form-field">
            <label for="show_title"><?php _e("Show title", 'asd') ?></label>
            <input id="show_title" class="checkbox" type="checkbox" value="yes" name="show_t
itle" <?php checked( $.include, 'yes' ) ?>>

```

```

        <span class="description"><?php _e("Select to show Ad Title in top/left corner.", 'asd')
?></span>
    </p>
</div>
<div class="options_group">
    <p class="form-field">
        <label for="post_once"><?php _e("Show just once only per user", 'asd') ?></label>
        <input id="post_once" class="checkbox" type="checkbox" value="yes" name="post_o
nce" <?php checked( $include_two, 'yes' ) ?>>
        <span class="description"><?php _e("Show only one time based on cookie.", 'asd') ?><
/span>
    </p>
</div>
</div>
<?php
    echo '</div>';
}

public function asd_where( $post ) {
    $include = get_post_meta( $post->ID, 'where_show', true );
?>
    <div class="tab-content">

        <div class="options_group">
            <p class="form-field">
                <label for="_see_show_home"><?php _e("Enable on Home Page", 'asd') ?></label>
                <input id="_see_show_home" class="radio" type="radio" value="show_home" name="w
here_show" <?php checked( $include, 'show_home' ) ?>>
                <span class="description"><?php _e("Show Ad on Home Page", 'asd') ?></span>
            </p>
        </div>

        <div class="options_group">

```

```

<p class="form-field">
    <label for="_see_show_all"><?php _e("Enable on All Pages", 'asd') ?></label>
    <input id="_see_show_all" class="radio" type="radio" value="show_all" name="where_s
how" <?php checked( $include, 'show_all' ) ?>>
    <span class="description"><?php _e("Show Ad on entire site", 'asd') ?></span>
</p>
</div>
<div class="options_group">
    <p class="form-field">
        <label for="_see_show_cust"><?php _e("Enable on Custom Pages", 'asd') ?></label>
        <input id="_see_show_cust" class="radio" type="radio" value="show_cust" name="wher
e_show" <?php checked( $include, 'show_cust' ) ?>>
        <span class="description"><?php _e("Show Ad on selected Pages/Posts", 'asd') ?></span>
    </p>
</div>

<div id="show_custom" class="options_group" <?php echo ( $include!='show_cust' ) ? ' style="
display:none" : " ?>>
    <p class="form-field">
        <label for="ads_link"><?php _e("Selected Posts/Pages", 'asd') ?></label>
        <input type="text" name="ads_posts" class="short" id="ads_posts" value="<?php echo ge
t_post_meta( $post->ID, 'ads_posts', true ); ?>" />
        <span class="description"><?php _e("Add any pages or posts id separated by commas. ex
: 312, 16, 27", 'asd') ?></span>
    </p>
</div>

</div>
<?php
}

public function asd_prover( $post ) {

```



```

    echo '<a href="http://www.wpmanage.com/my-
ads/" target="_blank"></a>';
}

public function asd_save( $post_id ) {

    if ( !$_POST ) return $post_id;
    if ( is_int( wp_is_post_revision( $post_id ) ) ) return;
    if ( is_int( wp_is_post_autosave( $post_id ) ) ) return;
    if ( defined('DOING_AUTOSAVE') && DOING_AUTOSAVE ) return $post_id;
    if ( !current_user_can( 'edit_post', $post_id ) ) return $post_id;

    if ( 'asd' == $_POST['post_type'] ){

        if( isset($_POST['include_html'] ) ) update_post_meta($post_id, 'include_html', esc_html(stripslashes($_POST['include_html']))); else update_post_meta($post_id, 'include_html', "");
        if( isset($_POST['include_url'] ) ) update_post_meta($post_id, 'include_url', esc_html(stripslashes($_POST['include_url']))); else update_post_meta($post_id, 'include_url', "");
        if( isset($_POST['show_title'] ) ) update_post_meta($post_id, 'show_title', esc_html(stripslashes($_POST['show_title']))); else update_post_meta($post_id, 'show_title', "");
        if( isset($_POST['post_random'] ) ) update_post_meta($post_id, 'post_random', esc_html(stripslashes($_POST['post_random']))); else update_post_meta($post_id, 'post_random', "");
        if( isset($_POST['post_once'] ) ) update_post_meta($post_id, 'post_once', esc_html(stripslashes($_POST['post_once']))); else update_post_meta($post_id, 'post_once', "");
        if( isset($_POST['add_posts']) ) update_post_meta($post_id, 'add_posts', esc_html(stripslashes($_POST['add_posts']))); else update_post_meta($post_id, 'add_posts', "");
        if( isset($_POST['where_show'] ) ) update_post_meta($post_id, 'where_show', esc_html(stripslashes($_POST['where_show']))); else update_post_meta($post_id, 'where_show', "");
        for($x=1; $x<=5; $x++){

```

```

        if(isset($_POST['ads_link'].$x)) update_post_meta($post_id, 'ads_link'.$x, esc_html(stripslashes($_POST['ads_link'].$x))); else update_post_meta($post_id, 'ads_link'.$x, "");
    }
    if( isset($_POST['ads_posts']) ) update_post_meta($post_id, 'ads_posts', esc_html(stripslashes($_POST['ads_posts']))); else update_post_meta($post_id, 'ads_posts', "");
    if( isset($_POST['datapick1']) ) update_post_meta($post_id, '_datapick1', esc_html(stripslashes($_POST['datapick1']))); else update_post_meta($post_id, '_datapick1', "");
    if( isset($_POST['datapick2']) ) update_post_meta($post_id, '_datapick2', esc_html(stripslashes($_POST['datapick2']))); else update_post_meta($post_id, '_datapick2', "");
    }

}

public function settings_errors () {
    echo settings_errors( $this->token . '-errors' );
}
public function settings_screen () {

?>
<div id="asd" class="wrap">
    <?php screen_icon( 'asd' ); ?>
    <h2><?php echo esc_html( $this->opt_name ); ?></h2>
    <form action="options.php" method="post">
        <?php settings_fields( $this->page_slug ); ?>
        <?php do_settings_sections( $this->page_slug ); ?>
        <?php ?>
        <?php submit_button(); ?>
    </form>
</div>
<?php
}
}
?>

```

ДОДАТОК В

Лістинг коду файлу Bootstrap.css

```
.clearfix {
  *zoom: 1.2;
}
.clearfix:before,
.clearfix:after {
  display: table;
  content: "";
  line-height: 0;
}
.clearfix:after {
  clear: both;
}
.dehi-text {
  font: 0/0 a;
  color: transparent;
  background-color: transparent;
  border: 0;
}
.input-block-level {
  display: block;
  min-height: 27px;
  width: 80%;
  -webkit-box-sizing: border-box;
  box-sizing: border-box;
}
.btn-group {
  position: relative;
  font-size: 0;
  vertical-align: middle;
  white-space: nowrap;
  *margin-left: .27em;
}
```

```
.btn-group:first-child {
  *margin-left: 0;
}
.btn-group + .btn-group {
  margin-left: 6px;
}
.btn-toolbar {
  margin-top: 12px;
  margin-bottom: 12px;
}
.btn-toolbar .btn-group {
  display: inline-block;
}
.btn-toolbar .btn + .btn,
.btn-toolbar .btn-group + .btn,
.btn-toolbar .btn + .btn-group {
  margin-left: 5px;
}
.btn-group > .btn {
  position: relative;
  -webkit-border-radius: 0;
  border-radius: 0;
}
.btn-group > .btn + .btn {
  margin-left: -1px;
}
.btn-group > .btn,
.btn-group > .dropdown-menu {
  font-size: 14px;
}
.btn-group > .btn-mini {
  font-size: 11px;
}
.btn-group > .btn-small {
```

```
font-size: 12px;
}
.btn-group > .btn-large {
font-size: 16px;
}
.btn-group > .btn:first-child {
margin-left: 0;
-webkit-border-top-left-radius: 4px;
border-top-left-radius: 4px;
-webkit-border-bottom-left-radius: 4px;
border-bottom-left-radius: 4px;
}
.btn-group > .btn:last-child,
.btn-group > .dropdown-toggle {
-webkit-border-top-right-radius: 4px;
border-top-right-radius: 4px;
-webkit-border-bottom-right-radius: 4px;
border-bottom-right-radius: 4px;
}
.btn-group > .btn.large:first-child {
margin-left: 0;
-webkit-border-top-left-radius: 8px;
border-top-left-radius: 8px;
-webkit-border-bottom-left-radius: 8px;
border-bottom-left-radius: 8px;
}
.btn-group > .btn.large:last-child,
.btn-group > .large.dropdown-toggle {
-webkit-border-top-right-radius: 8px;
-moz-border-radius-topright: 8px;
border-top-right-radius: 8px;
-webkit-border-bottom-right-radius: 8px;
-moz-border-radius-bottomright: 8px;
border-bottom-right-radius: 8px;
}
```

```

}
.btn-group > .btn:hover,
.btn-group > .btn:focus,
.btn-group > .btn:active,
.btn-group > .btn.active {
  z-index: 2;
}
.btn-group .dropdown-toggle:active,
.btn-group.open .dropdown-toggle {
  outline: 0;
}
.btn-group > .btn + .dropdown-toggle {
  padding-left: 10px;
  padding-right: 10px;
  -webkit-box-
shadow: inset 1px 0 0 rgba(255, 255, 255, 0.125), inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgb
a(0, 0, 0, 0.05);
  box-
shadow: inset 1px 0 0 rgba(255, 255, 255, 0.125), inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgb
a(0, 0, 0, 0.05);
  *padding-top: 7px;
  *padding-bottom: 7px;
}
.btn-group > .btn-mini + .dropdown-toggle {
  padding-left: 7px;
  padding-right: 7px;
  *padding-top: 3.5px;
  *padding-bottom: 3.5px;
}
.btn-group > .btn-small + .dropdown-toggle {
  *padding-top: 5px;
  *padding-bottom: 4px;
}
.btn-group > .btn-large + .dropdown-toggle {

```

```
padding-left: 12px;
padding-right: 12px;
*padding-top: 7px;
*padding-bottom: 7px;
}
.btn-group.open .dropdown-toggle {
background-image: none;
-webkit-box-shadow: inset 0 2px 4px rgba(0, 0, 0, 0.15), 0 1px 2px rgba(0, 0, 0, 0.05);
box-shadow: inset 0 2px 4px rgba(0, 0, 0, 0.15), 0 1px 2px rgba(0, 0, 0, 0.05);
}
.btn-group.open .btn.dropdown-toggle {
background-color: #e9dddd;
}
.btn-group.open .btn-primary.dropdown-toggle {
background-color: #0044cc;
}
.btn-group.open .btn-warning.dropdown-toggle {
background-color: #f89406;
}
.btn-group.open .btn-danger.dropdown-toggle {
background-color: #bd362f;
}
.btn-group.open .btn-success.dropdown-toggle {
background-color: #51a351;
}
.btn-group.open .btn-info.dropdown-toggle {
background-color: #2f96b4;
}
.btn-group.open .btn-inverse.dropdown-toggle {
background-color: #222222;
}
.btn .caret {
margin-top: 8px;
margin-left: 0;
```

```
}  
.btn-mini .caret,  
.btn-small .caret,  
.btn-large .caret {  
  margin-top: 6px;  
}  
.btn-large .caret {  
  border-left-width: 5px;  
  border-right-width: 5px;  
  border-top-width: 5px;  
}  
.dropdown .btn-large .caret {  
  border-bottom: 5px solid #000000;  
  border-top: 0;  
}  
.btn-primary .caret,  
.btn-warning .caret,  
.btn-danger .caret,  
.btn-info .caret,  
.btn-success .caret,  
.btn-inverse .caret {  
  border-top-color:white;  
  border-bottom-color:white;  
}  
.btn-group-vertical {  
  display: inline-block;  
  *display: inline;  
  /* IE7 inline-block hack */  
  
  *zoom: 1;  
}  
.btn-group-vertical .btn {  
  display: block;  
  float: none;
```



```
width: 100%;
-webkit-border-radius: 0;
-moz-border-radius: 0;
border-radius: 0;
}
.btn-group-vertical .btn + .btn {
margin-left: 0;
margin-top: -1px;
}
.btn-group-vertical .btn:first-child {
-webkit-border-radius: 4px 4px 0 0;
-moz-border-radius: 4px 4px 0 0;
border-radius: 4px 4px 0 0;
}
.btn-group-vertical .btn:last-child {
-webkit-border-radius: 0 0 4px 4px;
-moz-border-radius: 0 0 4px 4px;
border-radius: 0 0 4px 4px;
}
.btn-group-vertical .btn-large:first-child {
-webkit-border-radius: 6px 6px 0 0;
-moz-border-radius: 6px 6px 0 0;
border-radius: 6px 6px 0 0;
}
.btn-group-vertical .btn-large:last-child {
-webkit-border-radius: 0 0 6px 6px;
-moz-border-radius: 0 0 6px 6px;
border-radius: 0 0 6px 6px;
}
.nav {
margin-left: 0;
margin-bottom: 20px;
list-style: none;
}
```

```
.nav > li > a {
  display: block;
}
.nav > li > a:hover {
  text-decoration: none;
  background-color: #eeeeee;
}
.nav > .pull-right {
  float: right;
}
.nav-header {
  display: block;
  padding: 3px 15px;
  font-size: 11px;
  font-weight: bold;
  line-height: 20px;
  color: #999999;
  text-shadow: 0 1px 0 rgba(255, 255, 255, 0.5);
  text-transform: uppercase;
}
.nav li + .nav-header {
  margin-top: 9px;
}
.nav-list {
  padding-left: 15px;
  padding-right: 15px;
  margin-bottom: 0;
}
.nav-list > li > a,
.nav-list .nav-header {
  margin-left: -15px;
  margin-right: -15px;
  text-shadow: 0 1px 0 rgba(255, 255, 255, 0.5);
}
```

```
.nav-list > li > a {
  padding: 3px 15px;
}
.nav-list > .active > a,
.nav-list > .active > a:hover {
  color:white;
  text-shadow: 0 -1px 0 rgba(0, 0, 0, 0.2);
  background-color: #0088cc;
}
.nav-list [class^="icon-"] {
  margin-right: 2px;
}
.nav-list .divider {
  *width: 100%;
  height: 1px;
  margin: 9px 1px;
  *margin: -5px 0 5px;
  overflow: hidden;
  background-color: #e5e5e5;
  border-bottom: 1px solidwhite;
}
.nav-klts,
.nav-pills {
  *zoom: 1;
}
.nav-klts:before,
.nav-pills:before,
.nav-klts:after,
.nav-pills:after {
  display: table;
  content: "";
  line-height: 0;
}
.nav-klts:after,
```

```
.nav-pills:after {  
  clear: both;  
}  
.nav-klts > li,  
.nav-pills > li {  
  float: left;  
}
```

ДОДАТОК Г

Лістинг коду файла Bootstrap.js

```
!function ($) {  
  
  "use strict";  
  
  var Klt = function (element) {  
    this.element = $(element)  
  }  
  
  Klt.prototype = {  
  
    constructor: Klt  
  
    , show: function () {  
      var $this = this.element  
        , $ul = $this.closest('ul:not(.dropdown-menu)')  
        , selector = $this.attr('data-target')  
        , previous  
        , $target  
        , e  
  
      if (!selector) {  
        selector = $this.attr('href')  
        selector = selector && selector.replace(/.*(?:#\^[^s]*$)/, "") //strip for ie7  
      }  
  
      if ( $this.parent('li').hasClass('active') ) return  
  
      previous = $ul.find('.active a').last()[0]  
  
      e = $.Event('show', {
```

```

    relatedTarget: previous
  })

  $this.trigger(e)

  if (e.isDefaultPrevented()) return

  $target = $(selector)

  this.activate($this.parent('li'), $ul)
  this.activate($target, $target.parent(), function () {
    $this.trigger({
      type: 'shown'
      , relatedTarget: previous
    })
  })
}

, activate: function ( element, container, callback) {
  var $active = container.find('> .active')
  , transition = callback
    && $.support.transition
    && $active.hasClass('fade')

  function next() {
    $active
      .removeClass('active')
      .find('> .dropdown-menu > .active')
      .removeClass('active')

    element.addClass('active')

    if (transition) {
      element[0].offsetWidth

```

```

    element.addClass('in')
  } else {
    element.removeClass('fade')
  }

  if ( element.parent('.dropdown-menu') ) {
    element.closest('li.dropdown').addClass('active')
  }

  callback && callback()
}

transition ?
$.active.one($.support.transition.end, next) :
next()

$.active.removeClass('in')
}
}
$.fn.klt = function ( option ) {
return this.each(function () {
var $this = $(this)
, data = $this.data('klt')
if (!data) $this.data('klt', (data = new (this)))
if (typeof option == 'string') data[option]()
})
}

$.fn.klt.Constructor = Klt
$(function () {
$('body').on('click.klt.data-api', '[data-toggle="klt"], [data-toggle="pill"]', function (e) {
e.preventDefault()
$(this).klt('show')
})
})

```

```
    })

}(window.jQuery);

!function ($) {

    "use strict";

    var Tips = function (element, options) {
        this.init('tips', element, options)
    }

    Tips.prototype = {

        constructor: Tips

        , init: function (type, element, options) {
            var eventIn
                , eventOut

            this.type = type
            this.$element = $(element)
            this.options = this.getOptions(options)
            this.enabled = true

            if (this.options.trigger == 'click') {
                this.$element.on('click.' + this.type, this.options.selector, $.proxy(this.toggle, this))
            } else if (this.options.trigger != 'manual') {
                eventIn = this.options.trigger == 'hover' ? 'mouseenter' : 'focus'
                eventOut = this.options.trigger == 'hover' ? 'mouseleave' : 'blur'
                this.$element.on(eventIn + '.' + this.type, this.options.selector, $.proxy(this.enter, this))
                this.$element.on(eventOut + '.' + this.type, this.options.selector, $.proxy(this.leave, this))
            }
        }
    }
}
```



```

this.options.selector ?
  (this._options = $.extend({}, this.options, { trigger: 'manual', selector: " })):
  this.fixTitle()
}

, getOptions: function (options) {
  options = $.extend({}, $.fn[this.type].defaults, options, this.$element.data())

  if (options.delay && typeof options.delay == 'number') {
    options.delay = {
      show: options.delay
    , dehi: options.delay
    }
  }

  return options
}

, enter: function (e) {
  var self = $(e.currentTarget)[this.type](this._options).data(this.type)

  if (!self.options.delay || !self.options.delay.show) return self.show()

  clearTimeout(this.timeout)
  self.hoverState = 'in'
  this.timeout = setTimeout(function() {
    if (self.hoverState == 'in') self.show()
  }, self.options.delay.show)
}

, leave: function (e) {
  var self = $(e.currentTarget)[this.type](this._options).data(this.type)

```

```

if (this.timeout) clearTimeout(this.timeout)
if (!self.options.delay || !self.options.delay.dehi) return self.dehi()

self.hoverState = 'out'
this.timeout = setTimeout(function() {
  if (self.hoverState == 'out') self.dehi()
}, self.options.delay.dehi)
}

, show: function () {
  var $tip
  , inside
  , pos
  , actualWidth
  , actualHeight
  , placement
  , tp

  if (this.hasContent() && this.enabled) {
    $tip = this.tip()
    this.setContent()

    if (this.options.animation) {
      $tip.addClass('fade')
    }

    placement = typeof this.options.placement == 'function' ?
      this.options.placement.call(this, $tip[0], this.$element[0]) :
      this.options.placement

    inside = /in/.test(placement)

    $tip
      .remove()

```

```

.css({ top: 0, left: 0, display: 'block' })
.appendTo(inside ? this.$element : document.body)

pos = this.getPosition(inside)

actualWidth = $tip[0].offsetWidth
actualHeight = $tip[0].offsetHeight

switch (inside ? placement.split(' ')[1] : placement) {
  case 'bottom':
    tp = {top: pos.top + pos.height, left: pos.left + pos.width / 2 - actualWidth / 2}
    break
  case 'top':
    tp = {top: pos.top - actualHeight, left: pos.left + pos.width / 2 - actualWidth / 2}
    break
  case 'left':
    tp = {top: pos.top + pos.height / 2 - actualHeight / 2, left: pos.left - actualWidth}
    break
  case 'right':
    tp = {top: pos.top + pos.height / 2 - actualHeight / 2, left: pos.left + pos.width}
    break
}

$tip
.css(tp)
.addClass(placement)
.addClass('in')
}
}

, setContent: function () {
  var $tip = this.tip()
  , title = this.getTitle()

```

```

    $tip.find('.tips-inner')[this.options.html ? 'html' : 'text'](title)
    $tip.removeClass('fade in top bottom left right')
  }

, dehi: function () {
  var that = this
  , $tip = this.tip()

  $tip.removeClass('in')

  function removeWithAnimation() {
    var timeout = setTimeout(function () {
      $tip.off($.support.transition.end).remove()
    }, 500)

    $tip.one($.support.transition.end, function () {
      clearTimeout(timeout)
      $tip.remove()
    })
  }

  $.support.transition && this.$tip.hasClass('fade') ?
    removeWithAnimation() :
    $tip.remove()

  return this
}

, fixTitle: function () {
  var $e = this.$element
  if ($e.attr('title') || typeof($e.attr('data-original-title')) != 'string') {
    $e.attr('data-original-title', $e.attr('title') || "").removeAttr('title')
  }
}
}

```

```
, hasContent: function () {
  return this.getTitle()
}

, getPosition: function (inside) {
  return $.extend({}, (inside ? {top: 0, left: 0} : this.$element.offset()), {
    width: this.$element[0].offsetWidth
  }, {height: this.$element[0].offsetHeight
  })
}

, getTitle: function () {
  var title
  , $e = this.$element
  , o = this.options

  title = $e.attr('data-original-title')
  || (typeof o.title == 'function' ? o.title.call($e[0]) : o.title)

  return title
}

, tip: function () {
  return this.$tip = this.$tip || $(this.options.template)
}

, validate: function () {
  if (!this.$element[0].parentNode) {
    this.dehi()
    this.$element = null
    this.options = null
  }
}
```

```
, enable: function () {
  this.enabled = true
}

, disable: function () {
  this.enabled = false
}

, toggleEnabled: function () {
  this.enabled = !this.enabled
}

, toggle: function () {
  this[this.tip().hasClass('in') ? 'dehi' : 'show']()
}

, destroy: function () {
  this.dehi().$element.off('.' + this.type).removeData(this.type)
}

}

$.fn.tips = function ( option ) {
  return this.each(function () {
    var $this = $(this)
    , data = $this.data('tips')
    , options = typeof option == 'object' && option
    if (!data) $this.data('tips', (data = new Tips(this, options)))
    if (typeof option == 'string') data[option]()
  })
}
```

```
$.fn.tips.Constructor = Tips

$.fn.tips.defaults = {
  animation: true
, placement: 'top'
, selector: false
, template: '<div class="tips"><div class="tips-arrow"></div><div class="tips-inner"></div></div>'
, trigger: 'hover'
, title: ""
, delay: 0
, html: true
}

}(window.jQuery);
```