

**Київський національний торговельно-економічний
університет**

Кафедра комп'ютерних наук та інформаційних систем

ВИПУСКНИЙ КВАЛІФІКАЦІЙНИЙ ПРОЕКТ

на тему:

**«Розробка рекомендаційної Web-системи
інформаційного наповнення сайту для оптимального
відображення в пошукових системах»**

Студента 2 курсу, групи 7м
спеціальності
122 «Комп'ютерні науки»
спеціалізації

Кислюк Михайло
Сергійович

підпис студента

Науковий керівник
доктор фізико-математичних наук,
професор

Пурський Олег
Іванович

підпис керівника

Гарант освітньої програми
доктор фізико-математичних наук,
професор

Пурський Олег
Іванович

підпис керівника

Київ 2020

Київський національний торговельно-економічний університет

Факультет інформаційних технологій

Кафедра комп'ютерних наук та інформаційних систем

Спеціальність 122 «Комп'ютерні науки»

Зав. кафедри _____

Затверджую
Пурський О.І.
«5» грудня 2019р.

**Завдання
на випускний кваліфікаційний проєкт студенту**

Кислюку Михайлу Сергійовичу

(прізвище, ім'я, по батькові)

1. Тема випускної кваліфікаційної роботи (проєкту)

«Розробка рекомендаційної web-системи інформаційного наповнення сайту для оптимального відображення в пошукових системах»

Затверджена наказом ректора від «02» грудня 2019 р. № 4110

2. Строк здачі студентом закінченої роботи 05 листопада 2020 року

3. Цільова установка та вихідні дані до роботи

Мета роботи: програмна розробка рекомендаційної системи інформаційного наповнення сайту

Об'єкт дослідження: процеси інформаційного наповнення сайту

Предмет дослідження: інформаційні технології в пошукових системах

4. Перелік графічного матеріалу _____

5. Консультанти по роботі із зазначенням розділів, за якими здійснюється консультування:

Розділ	Консультант (прізвище, ініціали)	Підпис, дата	
		Завдання видав	Завдання прийняв
1	Пурський О.І.	5.12.2019 р.	5.12.2019 р..
2	Пурський О.І..	5.12.2019 р.	5.12.2019 р.
3	Пурський О.І.	5.12.2019 р.	5.12.2019 р.

6. Зміст випускної кваліфікаційної роботи (проекту) (перелік питань за кожним розділом)

ВСТУП

РОЗДІЛ 1. Аналіз механізмів залучення нових клієнтів за допомогою пошукових web-систем

1.1 Огляд можливих методів для збільшення кількості відвідувань сайту

1.2 Основні принципи необхідності ранжування у пошукових системах

1.3 Розподіл відвідувань сайту в залежності від позиції у пошукових системах

РОЗДІЛ 2. Огляд технік організації інформаційного наповнення сайту, що відображаються на ранжуванні у пошукових системах

2.1 Принципи побудови ієрархії сторінок на сайті

2.2 Основи побудови оптимізованого контенту на сайті

2.3 Рекомендації щільності ключових слів для кращого ранжування

РОЗДІЛ 3. Розробка системи для рекомендацій по наповненню сайту

3.1 Розробка моделі рекомендаційної системи

3.2 Специфіка програмної реалізації за допомогою фреймворку Django та Angular

3.3 Технологія використання рекомендаційної системи по наповненню сайту ВИСНОВКИ

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

ДОДАТОК

7. Календарний план виконання роботи

№ Пор.	Назва етапів випускної кваліфікаційної роботи	Строк виконання етапів роботи	
		За планом	фактично
1	2	3	4
1	Вибір теми випускної кваліфікаційної роботи	01.11.2019	01.11.2019
2	Розробка та затвердження завдання на випускну кваліфікаційну роботу	05.12.2019	05.12.2019
3	Вступ	01.06.2020	
4	РОЗДІЛ 1 Аналіз механізмів залучення нових клієнтів за допомогою пошукових web-систем	25.06.2020	
5	РОЗДІЛ 2. Огляд технік організації інформаційного наповнення сайту що відображаються на ранжуванні у пошукових системах	02.09.2020	
6	РОЗДІЛ 3. Розробка системи для рекомендацій по наповненню сайту	09.09.2020	
7	Висновки	21.10.2020	
8	Здача випускної кваліфікаційної роботи на кафедрі науковому керівнику	02.11.2020	
9	Попередній захист випускної кваліфікаційної роботи	05.11.2020	
11	Виправлення зауважень, зовнішнє рецензування випускної кваліфікаційної роботи	20.11.2020	
12	Представлення готової зшитої випускної кваліфікаційної роботи на кафедрі	22.11.2020	
13	Публічний захист випускної кваліфікаційної роботи	За розкладом роботи ЕК	

8. Дата видачі завдання «5» грудня 2019 р.

9. Керівник випускного кваліфікаційного проєкту Пурський О.І.
(прізвище, ініціали, підпис)

10. Гарант освітньої програми Пурський О.І.
(прізвище, ініціали, підпис)

11. Завдання прийняв до виконання студент-дипломник
Кислюк Михайло Сергійович
(прізвище, ініціали, підпис)

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There is no text or other markings on the paper.

« _____ » _____ 2020р.
(ніднuc, дама)

Випускна кваліфікаційна робота (проект) студента _____
(прізвище, ініціали)
може бути допущена до захисту в екзаменаційній комісії.

Завідувач кафедри _____ Пурський О.І.
(підпис, прізвище, ініціали)

« _____ » 2020 р.

АНОТАЦІЯ

Випускний кваліфікаційний проєкт присвячений розробці рекомендаційної web-системи інформаційного наповнення сайту для оптимального відображення в пошукових системах. Проведено теоретичний аналіз механізмів залучення нових клієнтів за допомогою пошукових систем. Представлені особливості організації інформаційного наповнення сайту, наведені переваги застосування SILO-архітектури. Запропоновано основні рекомендації з щільності ключових слів та їх розташування на сторінці web-сайту. Розроблена модель рекомендаційної web-системи з наповнення сайту.

Ключові слова: архітектура, ключові слова, контент, ранжування, пошукова система.

ANNOTATION

The graduated qualification work is devoted to the development of a recommended web-system of information content of the web-site for optimal display in search engines. Theoretically, the mechanisms of attracting new customers through search engines are presented. The peculiarities of the organization of information content of the site are presented, the advantages of application of SILO-architecture are given. The main recommendations on keyword density and their location on the web-site page are offered. A model of a recommended web-system for filling the site has been developed.

Key words: architecture, keywords, content, ranking, search engine.

ЗМІСТ

ВСТУП.....	9
РОЗДІЛ 1. АНАЛІЗ МЕХАНІЗМІВ ЗАЛУЧЕННЯ НОВИХ КЛІЄНТІВ ЗА ДОПОМОГОЮ ПОШУКОВИХ WEB-СИСТЕМ.....	12
1.1. Огляд можливих методів для збільшення кількості відвідувань сайту	12
1.2 Основні принципи необхідності ранжування у пошукових системах	22
1.3 Розподіл відвідувань сайту в залежності від позиції у пошукових системах	24
РОЗДІЛ 2. ОГЛЯД ТЕХНІК ОРГАНІЗАЦІЇ ІНФОРМАЦІЙНОГО НАПОВНЕННЯ САЙТУ, ЩО ВІДОБРАЖАЮТЬСЯ НА РАНЖУВАННІ У ПОШУКОВИХ СИСТЕМАХ.....	28
2.1 Принципи побудови ієрархії сторінок на сайті	28
2.2 Основи побудови оптимізованого контенту на сайті.....	33
2.3 Рекомендації щільності ключових слів для кращого ранжування	45
РОЗДІЛ 3. РОЗРОБКА СИСТЕМИ ДЛЯ РЕКОМЕНДАЦІЙ ПО НАПОВНЕННЮ САЙТУ.....	51
3.1 Розробка моделі рекомендаційної системи	51
3.2 Специфіка програмної реалізації за допомогою фреймворку Django та Angular	55
3.3 Технологія використання рекомендаційної системи по наповненню сайту	61
ВИСНОВКИ	63

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	65
----------------------------------	----

ДОДАТОК.....	69
--------------	----

ВСТУП

В теперішній час створюється і функціонує велика кількість різноманітних web-сайтів. Коли виконується пошук, пошукові системи переглядають свій індекс в пошуках високорелевантного контенту, а потім показують цей контент з метою вирішити запит шукача. Упорядкування результатів пошуку за релевантністю називається ранжируванням. Загалом, вважається, що чим вище рейтинг веб-сайту, тим вища його релевантність, і, при цьому в контексті пошукової системи, він найбільш вірогідно відповідає запиту. Можна заблокувати пошукові роботи для частини або всього сайту чи дати вказівку пошуковим системам уникати збереження певних сторінок в своєму індексі, хоча для цього повинні бути певні причини. Якщо хочеться, щоб контент знаходили пошуковики, треба спочатку переконатися, що web-сайт доступний для пошукових роботів і індексується.

Згідно зі статистикою, до 60% трафіку може бути отримано за рахунок звичайного пошуку у пошукових системах. Також статистичні дані показують, що більш ніж 90% відвідувань сайту з усієї кількості відвідувань приходить на першу пошукову сторінку. Це говорить про необхідність наявності процедури ранжування у пошукових системах. Існує значна кількість рекомендацій щодо оптимального наповнення web-сайту та його просування у пошукових мережах. Однак на сьогодні можна спостерігати і багато розходжень у таких рекомендаціях.

У зв'язку з вище зазначеними обставинами, **актуальність** обраної тематики дослідження визначається необхідністю розробки web-системи, яка б забезпечувала ефективне інформаційне наповнення web-сайту з метою оптимального відображення в пошукових системах.

Метою дослідження є розробка рекомендаційної web-системи інформаційного наповнення сайту для оптимального відображення в пошукових системах.

Для досягнення поставленої мети необхідно було вирішити наступні **завдання**:

- Виконати аналіз механізмів залучення нових клієнтів за допомогою пошукових web-систем;
- Дослідити принципи ранжування у пошукових системах;
- Визначити принципи побудови ієрархії сторінок на сайті;
- Здійснити огляд існуючих технік організації інформаційного наповнення сайту, що відображаються на ранжуванні у пошукових системах;
- Розробити модель рекомендаційної системи;
- Розробити web-систему для рекомендацій по наповненню сайту.
- Розробити технологію використання рекомендаційної web-системи.

Об'єкт дослідження: процеси інформаційного наповнення сайту.

Предмет дослідження: інформаційні технології в пошукових системах.

Методи дослідження: При написанні магістерської дисертації були застосовані методи аналізу, статистики, порівняння. Теоретичною основою є загальнонауковий аналітичний метод. Для практичного вирішення поставлених задач використовувалися такі методи:

- метод модульного проектування;
- методи об'єктно-орієнтованого програмування, для створення системи інформаційного наповнення web-сайту.

Наукова новизна одержаних результатів полягає в розробці

рекомендаційної web-системи інформаційного наповнення сайту, що забезпечує його оптимальне відображення в пошукових системах.

Практичне значення. Отримані результати можуть використовуватися при розробці сайту для представлення оптимізованого контенту з метою ефективного відображення сайту у пошукових системах.

Публікації. Результати дослідження опубліковано у збірнику наукових статей студентів, які здобувають освітній ступінь магістра за спеціалізацією «Комп'ютерні науки» КНТЕУ на тему: «Особливості SILO архітектури для проектування структури контенту сайту», 2020 р.

РОЗДІЛ 1.

АНАЛІЗ МЕХАНІЗМІВ ЗАЛУЧЕННЯ НОВИХ КЛІЄНТІВ ЗА ДОПОМОГОЮ ПОШУКОВИХ WEB-СИСТЕМ

1.1. Огляд можливих методів для збільшення кількості відвідувань сайту

В межах даного розділу дипломної роботи розглянемо основні механізми залучення клієнтів, які використовуються в нинішній час.

Трафік веб-сайту говорить, скільки користувачів відвідують сайт. Інструменти аналітики часто повідомляють як кількість переглядів сторінок, так і кількість унікальних переглядів. Перегляди сторінок - це загальна кількість разів, коли хто-небудь відвідував сайт, а унікальні перегляди сторінок говорять, скільки людей відвідали сайт - деякі кілька разів, а деякі тільки один раз [1].

Оскільки трафік показує як органічну популярність бренду, так і ефективність його маркетингових зусиль в Інтернеті, відвідуваність веб-сайту є критичним показником для будь-якого бізнесу. Таким чином, підвищення відвідуваності веб-сайтів як і раніше буде залишатися пріоритетом для зростання бізнесу в епоху цифрових технологій.

Розглянемо основні джерела трафіку.

Органічний трафік. Термін «органічний трафік» відноситься до відвідувачів, які потрапляють на веб-сайт через результати пошуку. Іншими словами, звичайні відвідувачі знаходять сайт, вводячи певні ключові слова в пошукові системи, такі як Google, Bing, Yahoo і інші, і потрапляють прямо на сайт [2].

Один з найпростіших і зручних способів збільшити органічний трафік веб-сайту - це регулярно публікувати високоякісний релевантний контент у вигляді блогів, тематичних досліджень, офіційних документів і

навіть відеоконтенту. Використання інших методів, таких як пошукова оптимізація, також має стати частиною стратегії зростання онлайн-трафіку.

Платний трафік. Платний трафік, як випливає з назви, являє собою трафік, отриманий за допомогою платних рекламних акцій, або через Google Рекламу, або через соціальні мережі і т. д. З безлічі платних механізмів, які компанія можна використовувати для отримання великого обсягу трафіку, можна виділити Pay Per Click (PPC) модель. Зазвичай це робиться через Google Ad Words. Інші відомі і часто використовувані типи включають банерну рекламу, рекламу з оплатою за придбання (PPA), ціну за тисячу показів (CPM).

Хоча більшість маркетологів припускають, що платна діяльність повинна складати лише невелику частину стратегії зростання трафіку, насправді вона може надати надзвичайно позитивний вплив на сайт (за умови, що вона буде ретельно спланована і виконана) з тієї причини, що платні акції гарантують результат [3].

На відміну від звичайних методів, платні механізми, здійснювані через авторитетні сайти, гарантують, що відвідувачі відвідують сайт і взаємодіють з його змістом. Можна також використовувати оптимізацію ключових слів для націлювання на певні сегменти аудиторії і залучення їх на свій сайт.

Реферальний трафік. Реферальний трафік - це відвідувачі, які приходять на сайт з джерел за межами Google. Коли відвідувач натискає гіперпосилання, щоб перейти на нову сторінку на іншому веб-сайті, Google Analytics відстежує це клацання як перехід на другий веб-сайт.

Вихідний сайт називається «сайтом реферера», тому що він відноситься до трафіку з одного сайту на інший. Реферальний трафік - це один з трьох типів статистичних даних, що відслідковуються Google Analytics.

Реферальний трафік також важливий для сайту, тому що він спрямовує можливих користувачів на сайт з інших сайтів, які можуть мати ту ж цільову групу. Відвідувачі вже довіряють рекомендаціям іншого сайту; отже, їм комфортно перенаправлятися на сайт.

Реферальний трафік також може приймати форму коду відстеження, розміщеного на інших веб-сайтах, включаючи рекламні банери, такі як AdWords, для включення коду реферала, пов'язаного з конкретною маркетинговою кампанією.

Маркетингові аналітики можуть легко перевірити список рефералів і краще визначити інтереси клієнтів, що приходять з цих сайтів. Можна також аналізувати це, і відповідно оптимізувати контент для споживання. Таким чином, можна легко перетворити цих відвідувачів в клієнтів.

Соціальний трафік. Соціальний трафік відноситься до обсягу трафіку, що надходить на сайт через соціальні мережі, такі як Facebook, Twitter, LinkedIn і т. д. Наприклад, якщо людина натискає на твіт або публікацію в Facebook, а потім потрапляє на сайт бренду, це вважається соціальним трафіком. Цей трафік також враховує як платний, так і органічний трафік.

Як і будь-яке інше джерело веб-трафіку, трафік соціальних мереж також можна використовувати для аналізу того, які сайти служать найбільш цінним джерелом трафіку для сайту.

Обчислюючи коефіцієнт конверсії цілі (GCR) [4] для кожного сайту соціальної мережі, можна оцінити, наскільки добре повідомлення знаходить відгук у аудиторії. Якщо повідомлення по суті і актуально для цільової аудиторії, цільові коефіцієнти конверсії напевно будуть високими. Тим часом, якщо коефіцієнт конверсії до цілі низький, необхідно переоцінити свою маркетингову стратегію в соціальних мережах, щоб залучити більше якісних лідів з сайтів соціальних мереж.

Електронний трафік. Інтернет-трафік, що направляється на сайт за допомогою маркетингових кампаній по електронній пошті, називається трафіком електронної пошти. Для бізнесу це один з найцінніших видів трафіку з тієї причини, що такий трафік виходить від ваших існуючих лідів і / або контактів.

Прямий трафік. Прямий трафік найчастіше є результатом введення користувачем URL-адреси в свій браузер або використання закладки для прямого доступу до веб-сайту. Прямі сеанси відбуваються кожен раз, коли Google Analytics не може визначити інше джерело або канал переходу. Цей фактор відрізняє прямий трафік від інших груп каналів за замовчуванням, таких як звичайний, реферальний, соціальний, електронний, платний і т. д. Однак прямий трафік також може складатися з відвідувачів, які потрапили на веб-сайт іншими способами.

Ось деякі з поширених випадків прямого трафіку:

- Доступ до немаркованого посилання з електронного листа;
- Клацання по посиланню з документа Microsoft Office або PDF;
- Доступ до сайту зі скороченого URL;
- Доступ до посилання з мобільного додатка соціальної мережі, такого як Facebook або Twitter в якості мобільних додатків, може не передавати інформацію про реферера;
- Випадок, коли користувач переходить на незахищений (HTTP) сайт за посиланням на захищеному (HTTPS) сайті. Це пов'язано з тим, що захищений веб-сайт не передає реферера на незахищений сайт. Наприклад, якщо користувач натискає посилання на test1.com, щоб перейти на test2.com, аналітика для example2.com покаже сеанс як прямий.

Доступ до веб-сайту зі звичайного пошуку може відображатися як прямий трафік через проблеми з браузером. Експеримент, проведений Groupm, показав, що до 60% прямого трафіку може бути отримано зі звичайного пошуку [5].

Інше. У цю категорію входять будь-які джерела, не віднесені до наведеного вище списку. Це може бути результатом будь-якої маркетингової кампанії, яку проводили на будь-якій іншій платформі для залучення трафіку. Будь-який некласифікований веб-трафік можна назвати іншим трафіком. Маркетингові аналітики можуть додатково вивчити це невизначене джерело трафіку і класифікувати його в будь-яку з перерахованих вище категорій.

Кожне джерело веб-трафіку є індикатором працездатності сайту. Великий обсяг реферального трафіку означає, що про веб-сайт часто говорять сторонні веб-сайти або сайти соціальних мереж. Хороша стратегія - додати ці метрики джерел трафіку на панель інструментів SEO, щоб глибше поринути в свої маркетингові зусилля [6].

На основі огляду літератури визначимо основні механізми підвищення кількості відвідувань сайту. Фактично є два способи залучити трафік на веб-сайт. Перший спосіб безкоштовний. Він включає в себе такі механізми, як пошукова оптимізація, створення блогу, маркетинг в соціальних мережах і так далі. Інший спосіб платний. Всім відомо, що платна реклама може залучити трафік на веб-сайт, але також може коштувати дорого [7].

Основні механізми збільшення кількості відвідувань:

1. Пошукова оптимізація фокусується на методі збільшення якості, так і кількості трафіку веб-сайту за допомогою звичайних результатів пошуку.

Якість трафіку дійсно важлива, тому що можна залучати всіх відвідувачів в світі на свій веб-сайт, але якщо вони не є цільовими відвідувачами, це марно.

Скажімо, наприклад, якщо відвідувачі приходять на сайт, тому що Google повідомляє їм, що компанія з виробництва ноутбуків, коли людина займається продажем чохлів для мобільних телефонів, такі відвідувачі не

приведуть до якісного трафіку. Замість цього потрібно залучати відвідувачів, які щиро зацікавлені в продуктах бренду, а не інакше.

Кількість трафіку. Зрозуміло, що великий обсяг трафіку завжди хороший для сайту. Але кількість тут нічого не означає без якості. Отже, якщо є потрібні люди / якісний трафік, що переходить зі сторінок результатів пошукових систем (SERP), більше трафіку - краще.

2. Органічні результати. Органічні результати пошуку, як правило, представляють собою списки результатів пошуку на сторінці результатів пошукової системи (SERP), які з'являються у верхній частині пошукової системи.

3. Реклама. Рекламу можна робити двома способами - платним і органічним. У той час як платна реклама відноситься до будь-якого виду реклами, здійснюваної через платні канали, органічна означає прямо протилежне. Сама по собі реклама допомагає підвищити популярність сайту у всесвітній павутині, заманюючи користувачів в Інтернеті, щоб перевірити сайт і поспілкуватися з ним. Це також забезпечує додаткову доступність веб-сайту як існуючим, так і потенційним клієнтам.

4. Соціальний медіа маркетинг Тільки створення відмінного контенту і надія на те, що люди його знайдуть і їм сподобаються, не спрацює. Необхідно бути «більш активними». Один з кращих і порівняно економічних способів збільшення відвідуваності веб-сайту - це використання каналів соціальних мереж для просування контенту. Є багато сайтів в соціальних мережах, і кожен має свою аудиторію і свою корисність.

Наприклад, Twitter в США ідеально підходить для коротких, швидких посилань. У LinkedIn є всі корпоративні і професійні аудиторії, де можна орієнтуватися на людей з певними призначеннями.

Якщо, наприклад, пропонуються продукти B2C, можна знайти велику популярність в соціальних мережах з великою кількістю зображень, таких як Pinterest і Instagram.

5. Відеомаркетинг. Відео - відмінний інструмент для збільшення відвідуваності сайту. Він не тільки здатний краще залучати відвідувачів, а й значною мірою знижує показник відмов.

Відеомаркетинг найкраще працює як частина більшої тематичної стратегії. Якщо користувач прагне привернути увагу глядачів і стати лідером в галузі, необхідно доповнити свій контент привабливими короткими відеороликами, що пояснюють контент, щоб підвищити залученість.

Крім того, рекомендується знайти правильні способи знайти баланс між письмовим і візуальним змістом. Рекомендується створити різні комбінації для різних типів продуктів або галузей і переконатися, що кожна з них красиво оформлена для отримання максимальної вигоди.

Звіт Small BizTrends показав, що відео мають на 1200% більше репоста в соціальних мережах в порівнянні з текстами і зображеннями. Завдяки репосту в соціальних мережах компанії знаходять нових клієнтів.

6. Веб-push-повідомлення Веб-push-повідомлення, інакше відомі як повідомлення браузера - інтерактивні фрагменти контенту, які стратегічно відправляються на пристрій користувача з метою привернути або повторно залучити відвідувача. Як правило, ці короткі інформативні фрагменти контенту відправляються тільки користувачам, які вибрали / підписалися на отримання push-повідомлень.

На відміну від інших носіїв, веб-push-повідомлення спеціально і ефективно використовуються для доставки обмеженого за часом контенту, щоб інформувати абонентів про останні зміни.

Деякі з популярних випадків перенацілювання користувачів за допомогою push-повідомлень включають відмову від перегляду,

перенацілювання споживачів, які покидають кошик, активацію бездіяльних клієнтів за допомогою цікавих пропозицій, утримання користувачів за допомогою персоналізованого контенту.

7. Інтернет-спільноти. Інтернет-спільноти - це групи людей, які діляться питаннями і відповідями, порадами, знаннями або історіями за спільними інтересами.

Хороший спосіб просування контенту, який може змінити ситуацію, - це ділитися релевантним контентом з сайту в онлайн-спільнотах. Ці спільноти можна знайти в соціальних мережах, таких як групи Facebook і LinkedIn.

Ці онлайн-спільноти являють собою ідеальну платформу для просування повідомлень в блогах. Інтернет-спільноти пропонують безліч переваг, наприклад:

- Підвищення впізнаваності: участь в онлайн-спільнотах дозволяє залучити в цільову аудиторію. Але що часто недооцінюється, так це той вплив, який участь може надати на впізнаваність.

Якщо власник сайту відчуває себе як експерта в галузі за допомогою засобів розповсюдження контенту, він збільшує трафік і збільшує кількість релевантних переглядів контенту і тим самим збільшує свої шанси на конверсію.

- Підвищення рівня залученості. Якщо використовуються онлайн-спільноти для правильного просування свого контенту, люди в групах будуть знати, які послуги або продукти пропонуються.

Відповідь на питання і відповіді, участь в чатах і відповідних форумах на цих платформах ще більше збільшує ступінь взаємодії з присутньою там аудиторією. В цілому це допомагає підвищити вашу впізнаваність, популярність і запам'ятовуваність.

8. Маркетинг впливу. Раніше компанії могли використовувати свій вплив для залучення потенційних, наприклад, покупців за допомогою платної реклами або будь-якої іншої маркетингової стратегії.

Покупці більше не чекають від компаній, щоб поділитися тим, що їм потрібно. Вони дивляться на інших людей. Люди, які впливають і мають досвід в певній галузі або продукті і цінуються в співтоваристві користувачів. Наприклад, блогери, відеоблогери YouTube, впливові особи в соціальних мережах і так далі.

Робота з такими впливовими особами - це розумний спосіб генерувати більше природних зворотних посилань кращої якості для сайту, тим самим покращуючи рейтинг сайту в пошуковій видачі.

9. Рекламна розсилка. Основна причина, по якій більшість маркетологів і раніше активно інвестують в електронний маркетинг, полягає в тому, що цей традиційний, але ефективний маркетинговий інструмент допомагає переконати потенційних клієнтів і перетворитися в лояльних клієнтів. Електронний маркетинг також є недорогим і дуже ефективним способом привернути увагу клієнтів і потенційних клієнтів.

Крім того, маркетингові кампанії по електронній пошті можна розумно використовувати для збільшення відвідуваності веб-сайту і підвищення рівня взаємодії.

Якщо необхідно поліпшити видимість свого веб-сайту, рекомендується інтегрувати маркетинг в соціальних мережах з кампаніями по електронній пошті.

Нижче наведені деякі практичні ідеї щодо залучення більшої кількості відвідувачів на сайт:

1. Видалення неякісного контенту зі свого веб-сайту. Блоги - це дуже хороший спосіб привернути аудиторію і збільшити веб-трафік. Однак люди зазвичай не беруть до уваги те, що також дуже важливо періодично

оновлювати свої блоги та інші інформаційні ресурси. Особливо необхідно оновлювати будь-які дані і статистику, яким можливо, поділилися в блозі.

2. Оптимізація сайту для мобільного SEO. SEO для мобільних пристроїв - це практика оптимізації мобільного веб-сайту і його зручності для користувачів. Цей метод також включає в себе забезпечення доступності ресурсів сайту для павуків пошукових систем.

58% всіх пошукових запитів в Google тепер виконуються з мобільних пристроїв [8]. Кількість пошукових запитів з мобільних пристроїв росте більше, ніж будь-коли, і можна з упевненістю сказати, що майбутнє пошуку за мобільними пристроями. І саме тому Google переглядає весь свій алгоритм, щоб зосередитися на мобільному пошуку. Індекс Google, орієнтований на мобільні пристрої, ранжує результати пошуку тільки на основі мобільної версії сторінки.

Інтернет-користувачі люблять ділитися корисним і унікальним контентом, таким як повідомлення в блогах, відео, зображення і інфографіка. Таким чином, чим більше контенту буде розміщено, тим більше буде можливостей поділитися своїм контентом, що створює зворотні посилання для сайту - безумовно, перевага SEO.

3. Більшість систем управління контентом пропонують базову аналітику для збору даних. Рекомендується отримати цю інформацію з кількісних інструментів, таких як Google Analytics, Це дає повну картину веб-трафіку і контенту.

В якості альтернативи можна розглянути будь-який інший інструмент, доступний на ринку. Кожен інструмент аналітики вимагає різного рівня знань для використання, хоча менш складні варіанти будуть більш зручними для використання в базових аналітичних цілях.

1.2 Основні принципи необхідності ранжування у пошукових системах

Згідно зі статистикою, до 60% трафіку може бути отримано зі звичайного пошуку у пошукових системах.

Рейтинг в пошукових системах не випадковий. Так, багато людей використовують Google для пошуку найрізноманітніших товарів по всьому світу. Google використовує унікальний алгоритм, що зберігається в секреті. Проте, існує кілька зручних для пошукових систем і навіть рекомендованих Google тактик, які компанії можуть використовувати, щоб підтримувати актуальність в Інтернеті і залишатися на вершині результатів пошуку по зв'язаних темах.

Це не тільки в кінцевому підсумку призводить до конверсії, а й створює онлайн-репутацію компаній як надійних джерел інформації, так і постачальників якісних продуктів або контенту [9].

Наприклад, такі компанії, як Slingshot SEO, допомагають компаніям реалізувати свій потенціал в рейтингу, не тільки аналізуючи поточну ситуацію з пошуком, але і надаючи інструменти, які необхідні компаніям для підвищення релевантності і авторитету в Інтернеті. Коли є така компанія, як Slingshot, яка допоможе отримати більш високий рейтинг, це зробить нескінченний процес набагато простіше і приємніше.

З появою різних способів пошуку, від пошуку в соціальних мережах до мобільних додатків, створених для пошуку, важливо, щоб компанії використовували всі доступні варіанти. Бути поміченим в Інтернеті - це маркетинговий інструмент не тільки через конверсію споживачів. Доступність в Інтернеті також показує, що компанія досить новаторська, щоб змінюватися відповідно до очікувань і потреб споживачів, зацікавлена в тому, щоб залишатися в центрі уваги споживачів, легко доступна для зворотного зв'язку і є експертом у відповідних областях.

Ранжування - це сортування пошуковою машиною списку результатів користувачеві згідно з внутрішніми алгоритмами, метою яких є релевантність пошуковому запиту.

Ранжування працює за наступною схемою [10]:

- Відправлення запиту користувача в пошуку;
- "Розуміння" запиту;
- Вивід максимально відповідних результатів за запитом;
- Фільтрація результатів від дублів і непотрібних результатів;
- Сортування згідно з алгоритмом;
- Вивід результату на сторінці пошуковика.

Завдання пошукової системи - обробляти запити користувача (у вигляді ключових слів) і пропонувати правильні відповіді (у вигляді видачі). Враховуючи це, пошукові системи приводять клієнтів, які дійсно зацікавлені у продукті, саме тому користувачі довіряють найбільше пошуковим системам, що, у свою чергу, приносить прибуток для сайту, який знаходиться зверху списку у системі.

Розпишемо короткий чек-лист, як потрібно створити та оптимізувати свій сайт для пошуку.

Якщо коротко, стратегія у великих пошукових систем особливо не відрізняється від трьох базових постулатів:

1. Актуальність інформації. Необхідно намагатися підтримувати інформацію «в тонусі»: будь то вартість товару, його опис і наявність, стаття-інструкція з актуальними скриншотами або стаття в блозі з активними коментарями.

2. Повнота інформації. Необхідно намагатися розділяти інформацію по темах і давати кожній сторінці свою назву - заголовок Title. Чим краще структурована інформація, уникаючи її дублювання, тим краще для сайту.

3. Популярність інформації. Контент - ніщо без просування. Необхідно посилятися на сайт з інших джерел: місцевих ЗМІ, тематичних ресурсів, трафікових майданчиків. Важливо, щоб посилання також мало свій трафік.

Підвищення рейтингу в SERP (сторінці результатів пошукової системи) не тільки підвищує здатність бізнесу генерувати трафік і потенційних клієнтів, але і збільшує ступінь, в якій аудиторія вважає бізнес справжнім [11].

Дані показують, що відвідувачі мережі думають, що перші кілька веб-сайтів, які займають верхні рядки видачі, відносяться до числа найбільш автентичних підприємств в своїй галузі. По суті, це психологічна гра, яка дає перевагу.

Але бути на вершині може бути досить складно, якщо є величезні конкуренти. Таким чином, незалежно від того, чи є ви великим або невеликим місцевим бізнесом, більш високий рейтинг, безсумнівно, підвищить достовірність вашого бізнесу.

1.3 Розподіл відвідувань сайту в залежності від позиції у пошукових системах

Google показує 10 результатів на першій сторінці, коли хтось шукає будь-які конкретні терміни. Результати засновані на ряді чинників, таких як швидкість сайту, зворотні посилання, показник відмов і безліч інших чинників.

Розподіл відвідувань сайту в залежності від позиції у пошукових системах представлені на рис. 1.1.

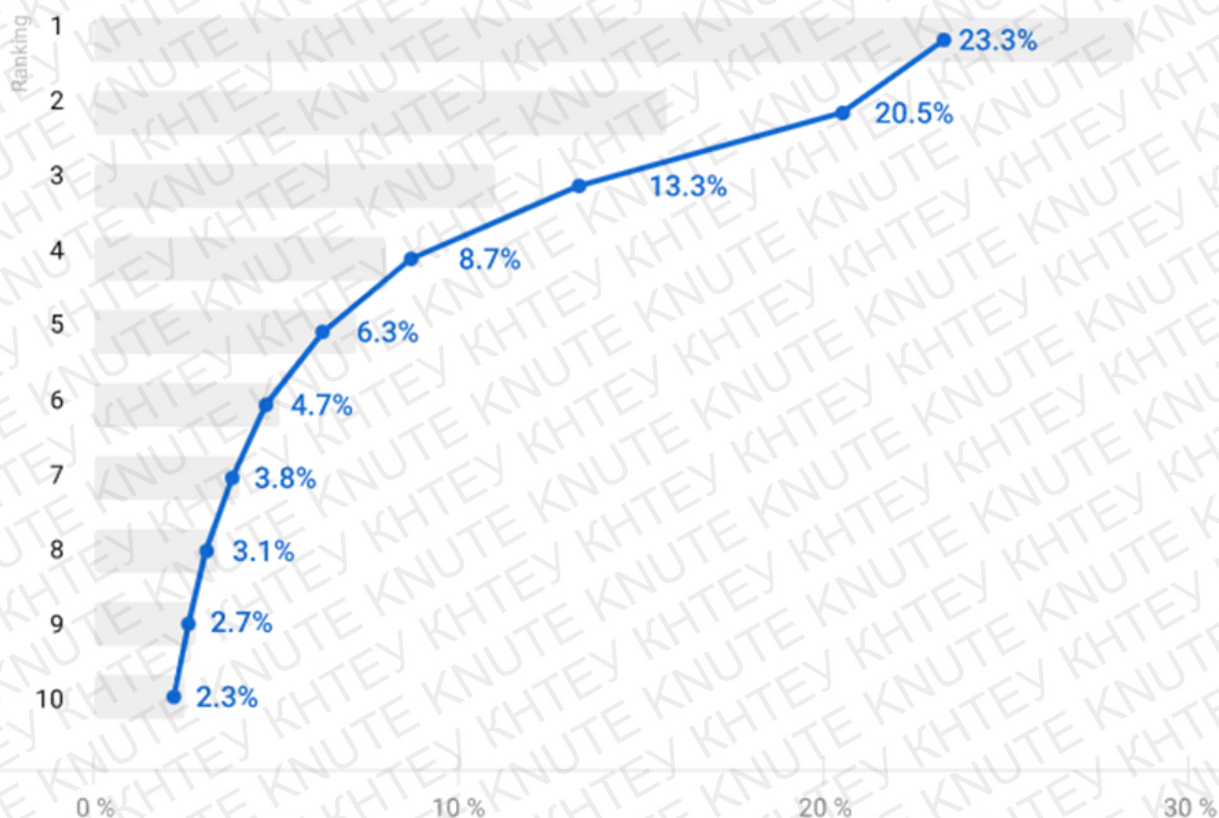


Рис. 1.1. Розподіл відвідувань сайту в залежності від позиції у пошуковій системі Google

Одна з інших провідних пошукових систем після Google, Bing, повідомляє, що веб-сайти, що знаходяться на вершині, отримують 32% трафіку; другий отримує 21%, а третій - тільки 8%. Таким чином, це ясно показує, чому сайти повинні бути на вершині.

Розподіл відвідувань сайту в залежності від позиції у пошуковій системі Yahoo представлений у вигляді табл. 1.1.

Таблиця 1.1. Розподіл відвідувань сайту у пошуковій системі Yahoo

Позиція	Процент відвідувань, %
1	32,5
2	17,6

3	11,4
4	8,1
5	6,1
6	4,4
7	3,5
8	3,1
9	2,6
10	2,4

Статистичні результати [12] також показали значуще падіння трафіку на сторінках 1 на сторінці 2. Результати сторінок 1 зібрали 92 відсотки всього трафіку від середнього пошуку, при цьому трафік перших двох сторінок зібрав 95 відсотків (рис. 1.2).

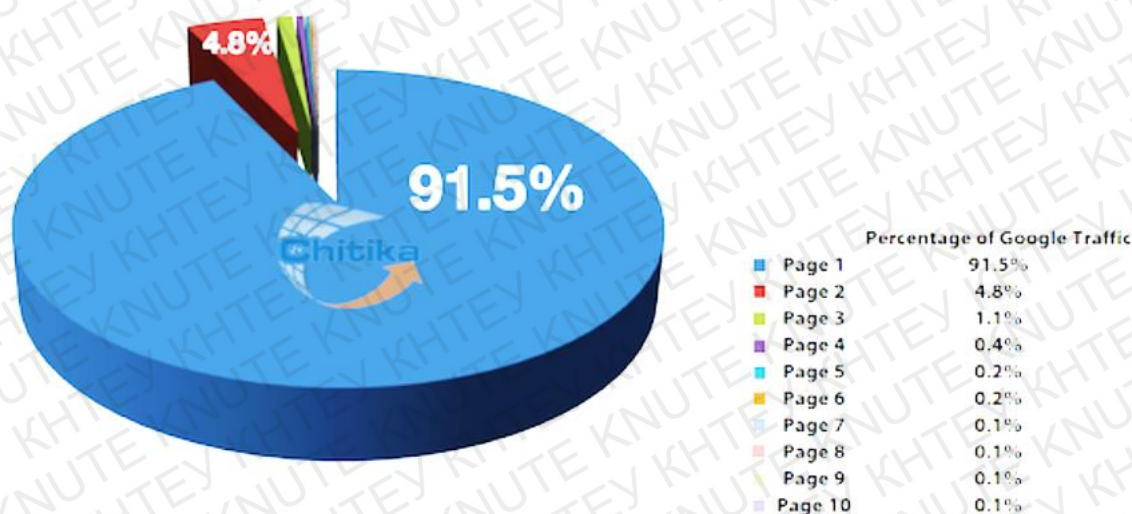


Рис. 1.2. Статистичний розподіл кількості відвідувань між сторінками Google

Представлені статистичні дані показують, що близько 91% відвідувань сайту з усієї кількості відвідувань приходить на першу пошукову сторінку. Це говорить про необхідність в ранжуванні у пошукових системах вище.

Отже, якщо веб-сайт стане видимим нагорі, це буде означати більшу автентичність і довіру, які є очевидними елементами для більшої кількості потенційних клієнтів і трафіку. А такі користувачі є найбільш необхідними.

РОЗДІЛ 2.

ОГЛЯД ТЕХНІК ОРГАНІЗАЦІЇ ІНФОРМАЦІЙНОГО НАПОВНЕННЯ САЙТУ, ЩО ВІДОБРАЖАЮТЬСЯ НА РАНЖУВАННІ У ПОШУКОВИХ СИСТЕМАХ

2.1 Принципи побудови ієрархії сторінок на сайті

В межах даного розділу роботи розглянемо основні техніки організації інформаційного наповнення сайту.

Для пошукової оптимізації часто досить внести невеликі зміни в деякі розділи сайту. Кожна таке зміна може здатися незначною, але в комплексі всі зміни можуть помітно підвищити зручність використання сайту і ефективність результатів пошуку [13].

Щоб зробити сайт зручним для відвідувачів, а також поліпшити його позицію в результатах пошуку, його потрібно оптимізувати. Пошукова оптимізація допомагає правильно інтерпретувати контент і представляти його в пошуку.

Для огляду технік організації інформаційного наповнення сайту варто спочатку розглянути принципи побудови ієрархії сторінок на сайті.

Інформаційна архітектура допускає організацію контенту як в плоскій, так і в глибокій ієрархії. Обидва способи мають свої переваги і недоліки.

Практично будь-який веб-ресурс, який має в своїй інформаційній архітектурі більш ніж пару-трійку сторінок, використовує якусь структуру для організації контенту. Найбільш поширена структура виглядає так: сторінки класифікуються в групи (категорії), часто в цих групах виділяються підгрупи [14].

Те, як саме буде згруповано контент, в подальшому може мати наслідки: в залежності від того, в яку структуру буде організований контент, сайт буде зручний (або незручний) для відвідувачів.

Для аналізу того, як структура буде працювати, розглянемо ілюстрацію типів ієрархій (рис. 2.1)

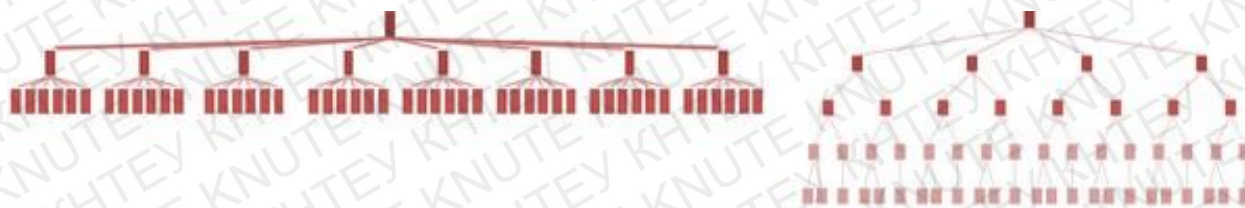


Рис. 2.1. Типи ієрархій

Розглянемо ці 2 структури: кожна з них сумарно містить один і той же обсяг інформації і показує абсолютно логічний спосіб організації контенту для веб-сайту. Проте досвід кінцевого користувача від взаємодії з кожною з цих 2 ієрархій - навіть якщо вони містять таку саму інформацію - буде радикально відрізнятися.

Зліва: плоска ієрархія сайту з декількома нечисленними рівнями по вертикалі. Справа: глибока ієрархія сайту, що володіє тією ж інформацією в більшій кількості рівнів.

Ієрархії обох веб-ресурсів починаються у верхній частині з однієї сторінки (головної / домашньої), але інформація на сторінках нижченаведеного рівня організована зовсім по-різному: сайт зліва має 8 основних категорій, сайт праворуч - тільки 4.

Пошуковим системам потрібен окремий URL для кожного фрагмента контенту, щоб мати можливість просканувати, проіндексувати його і показати його користувачам. Для різних фрагментів контенту (наприклад, для різних товарів або для регіональних варіантів і перекладів на інші мови) слід використовувати різні URL, щоб в результатах пошуку не виникало плутанини.

URL зазвичай складається декількох частин:

протокол: // ім'я_хоста / шлях / назва_файлу? параметр # якір

приклад:

<https://www.example.com/RunningShoes/Womens.htm?size=8#info>

Наприклад, Google рекомендує використовувати протокол `https: //` на всіх сайтах, де це можливо.

Ім'я хоста показує, де розміщується сайт, і зазвичай збігається з доменним ім'ям, яке використовується в адресі електронної пошти. Google відрізняє адреси, що починаються з `www` від інших адрес (наприклад, `www.example.com` і `example.com`). Коли додається свій сайт в Search Console, рекомендується додавати варіанти URL з `http: //` і `https: //`, а також з `www` і без `www`.

Шлях, ім'я файлу і параметр визначають, який контент покаже сервер. Ці частини чутливі до регістру, тому `FILE` і `file` дають різні URL. Ім'я хоста і протокол до регістру нечутливі.

Якір (в нашому випадку - `#info`) зазвичай вказує на частину всередині сторінки. Пошукові системи зазвичай їх ігнорують, тому що контент сторінки не залежить від використовуваних на ній якорів.

В кінці URL головної сторінки іноді зустрічається коса риска. Її наявність не грає ролі, бо як з нею, так і без неї це посилання на один і той же контент (`https://example.com/` - те ж саме, що `https://example.com`). У випадку зі шляхами і іменами файлів коса риска дає різні URL, наприклад `https://example.com/fish` посилається на файл, а `https://example.com/fish/` - на каталог.

Навігація на сайті важлива, оскільки вона дозволяє користувачам швидко знайти те, що їм потрібно. Також структура навігації допомагає пошуковим системам зрозуміти, який контент найбільш важливий. Наприклад, у результатах пошуку Google зазвичай представлені описи сторінок, однак пошуковій системі також важливо знати, яке місце займає сторінка в загальній структурі сайту.

На кожному сайті є головна сторінка, з якої зазвичай починається навігація по сайту і яку, отже, відвідують частіше, ніж інші сторінки сайту. Якщо у сайту досить складна структура, потрібно продумати, як

відвідувачі будуть переходити з головної сторінки до потрібного їм контенту.

Можливо, слід додати сторінку з їх переліком та описом, наприклад таким чином: головна сторінка> список сторінок по одній темі> тема. Якщо багато товарів, можливо, має сенс розбити їх на категорії і перерахувати ці категорії на окремих сторінках.

Як вже зазначалося, чим краще структура сайту, тим вищі шанси на більш високий рейтинг в пошукових системах. У кожного веб-сайту є деяка «структура». Це може бути сувора і обтічна структура, або це може бути безладний набір сторінок [15].

Перелічимо основні рекомендації з налаштування правильної архітектури веб-сайту [16]:

1. Не змушувати користувачів занадто багато думати. У веб-сайту з труднощами для навігації буде високий показник відмов - користувачі не хочуть витратити час на пошуки інформації на сайті. Якщо вони це зроблять, вони просто підуть. Необхідно забезпечити інтуїтивно зрозумілий веб-інтерфейс.

Наприклад, якщо користувачі натискають на вкладку «Маркетинг по електронній пошті» на головній сторінці блогу, вони очікують, що будуть перенаправлені на список повідомлень електронного маркетингу. На цій сторінці також необхідно створити простий шлях для переходу до домашньої сторінки блогу і домашньої сторінки сайту.

2. Змодельювати архітектуру свого веб-сайту за зразком кращих «гравців» у галузі. Клієнти звикли до архітектури веб-сайтів великих брендів галузі, тому, якщо проходить, наприклад, керування магазином електронної комерції, необхідно проаналізувати, як Amazon структурує їх веб-сайт і наслідує їх. Таким чином веб-сайт буде здаватися більш знайомим і, в свою чергу, з ним буде легше орієнтуватися.

3. Зробити сайт послідовним. Формат навігації веб-сайту, принципи дизайну та відображення посилань повинні слідувати єдиним шаблоном. Збереження цих елементів такими ж дозволить користувачам довше залишатися на сайті, тому що їм буде простіше швидко переходити на нові сторінки і натискати посилання.

4. Внутрішні посилання повинні мати сенс. Внутрішні посилання повинні направляти користувачів на інший релевантний і корисний контент. Крім того, коли користувачі стикаються з внутрішнім посиланням на веб-сайті, вони повинні відразу розуміти, на який фрагмент контенту буде направлено і чому цей контент пов'язаний з веб-сторінкою, на якій вони зараз знаходяться.

Тим не менш, необхідно дотримуватися одного внутрішнього застереження щодо посилань: Не вставляти ключові слова в якірний текст посилання. Google був свідком того, як люди роками вставляли ключові слова в такого роду текст своїх внутрішніх посилань, намагаючись перевершити їх алгоритм. Але насправді пошукова система створює особливі алгоритми для «покарання» за таку поведінку.

Також корисно мати надійну сторінку карти сайту для нижнього колонтитула або навігації верхнього рівня. Це допомагає пошуковим системам і користувачам знаходити сторінки на сайті набагато швидше і простіше.

5. Користувач повинен отримати доступ до будь-якої зі сторінок сайту за 3-4 кліка. Навіть якщо на веб-сайті мільйон сторінок, архітектура повинна дозволяти користувачам починати з головної сторінки і потрапляти на будь-яку сторінку за три-чотири кліка.

Для цього можна систему навігації верхнього рівня, яка буде направляти користувачів до всіх основних категорій сайту. Потім необхідно переконатися, що з кожної зі сторінок основних категорій веб-сайту вони можуть перейти на всі сторінки підкатегорій.

2.2 Основи побудови оптимізованого контенту на сайті

Кожен успішний веб-сайт повинен дозволяти користувачам легко і ефективно знаходити інформацію, яку вони шукають, тому вкрай важливо розмістити кожен елемент саме там, де аудиторія очікує його знайти. Для великих веб-сайтів з сотнями або тисячами сторінок це означає розуміння того, як згрупувати схожі сторінки за темами [18].

За останнє десятиліття Інтернет став місцем, де споживачі приймають рішення про покупку і купують товари.

Глобальні продажі електронної комерції зросли до більш ніж 3,46 трильйона доларів в 2019 році, що становить 16,4% від загальної роздрібної виручки в усьому світі.

Інтернет також став місцем, куди можна знайти інформацію про місцеві підприємства (46% від загального числа пошукових запитів в Google мають локальний характер, а 78% місцевих мобільних пошукових запитів приводять до покупок в офлайн-магазинах), а також про компанії-розробники програмного забезпечення, B2B підприємства і багато іншого.

Інтернет бере на себе роль «запитати друга» у всіх галузях. Таким чином, правильно позиціонуючи свій бізнес, можна постійно перетворювати людей, які раніше ніколи не чули про компанію, в клієнтів.

Якщо не відбувається оптимізація контенту, не має значення, скільки людей шукають терміни, що відносяться до бізнесу. Сайт не буде доступний в результатах.

Але коли проходить оптимізація пошуку, сайт починає виходити на верхні позиції. Споживачі будуть шукати релевантні терміни, і сайт завжди буде відображатися.

Освоївши пошукову оптимізацію (SEO), можна буде генерувати цільовий трафік від зацікавлених покупців.

На фундаментальному рівні структура веб-сайту повинна відповідати таким критеріям:

- Допомогти кожному новому відвідувачу легко і швидко зрозуміти інформацію, що міститься на сайті;
- Переконатися, що відвідувачі можуть легко і ефективно переміщатися за допомогою інтуїтивно зрозумілої навігації.

Добре структуровані веб-сайти, як правило, будуть мати низький показник відмов, тривалий час перебування (час перебування відвідувачів на сайті), хороший рейтинг кліків (CTR) і будуть ресурсом, з яким аудиторія активно хоче проводити час, взаємодіючи. Інтуїтивно зрозумілий інтерфейс - невід'ємна частина будь-якого хорошого веб-сайту, і все це починається з правильної структури і архітектури сайту [19].

Пошукові системи завжди шукають релевантність при організації сторінок результатів за запитами користувачів, і, оскільки більшість веб-сайтів містять масив сторінок, які не завжди можуть мати чітку тему, практика ефективного поділу допоможе прояснити релевантність і підвищити загальний рейтинг в довгостроковій перспективі [20].

Перше, що необхідно зрозуміти, це те, що між вмістом і оптимізацією існує велика різниця, і кожен термін застосовується і звертається до двох абсолютно різних аудиторій. Контент, наприклад, націлений на читачів - відмінний контент приверне читачів, підтримає їх інтерес і спонукає їх інвестувати в продукт або послугу.

Оптимізація, з іншого боку, має мало спільного з читачами і замість цього призначена для машин або роботів пошукових систем, які використовують текст для поліпшення SEO-рейтингу. Таким чином, неоптимізований контент може залишитися непоміченим пошуковою системою і, отже, непрочитаним масами [21].

По суті, оптимізація контенту або SEO (пошукова оптимізація) - це процес оптимізації контенту, щоб зробити його більш помітним в мережі. Роботи пошукових систем будуть ранжувати високооптимізований контент на сторінці пошукової системи вище, ніж неоптимізований контент.

Оптимізація веб-сайту включає безліч нюансів, і роботи пошукових систем зважають всі, від контенту до HTML і зворотних посилань.

Отже, одним з найбільш важливих аспектів оптимізації веб-сайтів є пошукова оптимізація. Цей метод не тільки спрямований на те, щоб різні сторінки веб-сайту займали високі позиції в пошуковій видачі за певними ключовими словами, а й дозволяє потенційним клієнтам знайти ваш найпростішим способом. Інший важливий аспект - оптимізація на сторінці. Цей метод гарантує, що потенційні клієнти, які потрапляють на веб-сайт, матимуть кращий користувацький досвід, змушуючи їх зробити бажану дію і перетворитися в ліда.

Таким чином, оптимізуючи свій веб-сайт, можна не тільки вийти на невикористаний ринок найбільш ефективним і дієвим чином (без оплати реклами), але і відкрити двері для більшої кількості конверсій і збільшення доходів [22].

Актуальність оптимізації полягає в тому, що вона для реальних людей допоможе завоювати довіру відвідувачів.

Цілісний підхід до оптимізації веб-сайту поєднує в собі різні дисципліни, щоб веб-сайт працював ідеально в усіх областях:

- SEO;
- копірайтинг;
- аналітика;
- UX-дизайн (фронтенд);
- Веб-розробка (Backend);
- CRO / Оптимізація цільової сторінки.

Уже недостатньо мати веб-сайт, який відмінно виглядає і працює на ноутбуках і настільних комп'ютерах. Щоб домогтися успіху на онлайн-ринку, також необхідно зосередитися на мобільних можливостях сайту.

Більша частина всього трафіку, як вже зазначалося, припадає на мобільний. Не кажучи вже про те, що більшість пошукових запитів в

Google зараз виконується на смартфонах. У третьому кварталі 2019 року 64% пошукових запитів в Google виконувалися з мобільних пристроїв.

Через це Google вже переключився на індексацію мобільних пристроїв, при якій вони в першу чергу індексують і ранжують мобільні сторінки. Таким чином, будь-які проблеми можуть коштувати не тільки потенційних мобільних конверсій, а й можливості отримати високий рейтинг по релевантним пошуковим запитам.

Отже, перше, що рекомендується зробити при оптимізації, це запустити базовий тест на зручність використання мобільних пристроїв, наприклад, Google Mobile-Friendly Test (рис. 2.2).

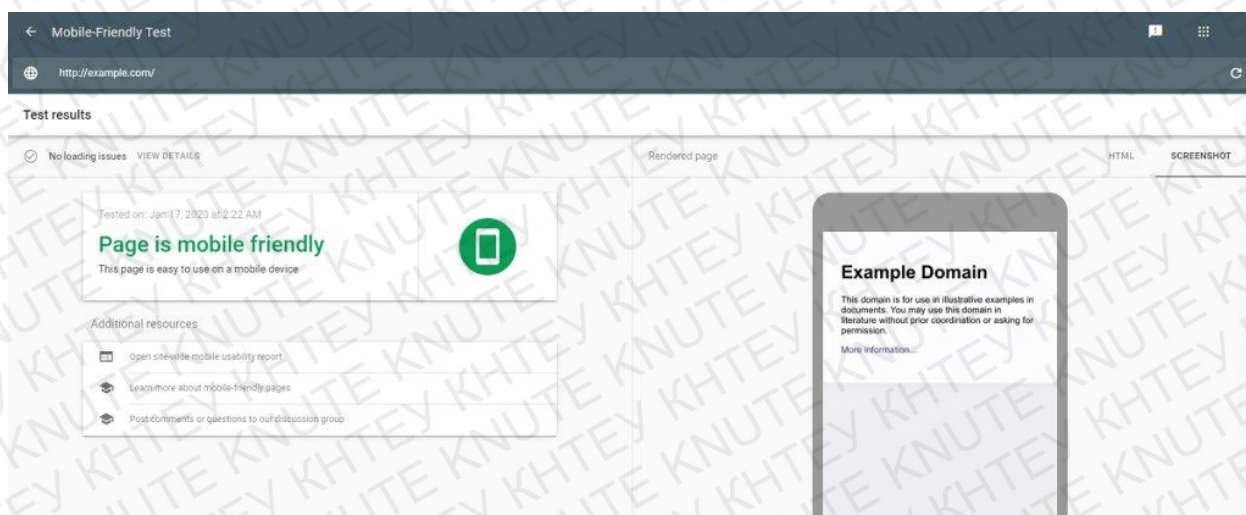


Рис. 2.2. Тест на зручність використання мобільних пристроїв

Необхідно переконатися, що:

- На мобільних пристроях не відображаються спливаючі вікна і міжсторінкові оголошення;
- Сайт завантажується швидко / правильно;
- Текст легко читається;
- Весь вміст видно;
- Перегляд ескізів і графіка як і раніше читані;
- На сайті легко орієнтуватися.

Однією з основних стратегій просування веб-сайту є безпосередня оптимізація контенту. Оптимізація контенту - це процес, в якому веб-сторінка і її контент оптимізуються, щоб стати більш привабливими, корисними й ефективними для користувачів. Ці процеси зазвичай, містять виправлення і поліпшення технічних характеристик (наприклад, швидкості сторінки) і копіювання контенту, щоб він працював і краще ранжувався в пошукових системах [23].

Конкуренція за увагу піднялася на новий рівень. З огляду на цей факт, величезна кількість видавців з різних вертикалей вже підтримали ідею просування набагато більшої кількості «відмінного контенту» з неухильно зростаючими темпами.

Хороший контент не такий гарний, коли він погано оптимізований. А оптимізація контенту, щоб зробити його більш конкурентоспроможним для пошуку, за ці роки зазнала величезних змін.

Оптимізація контенту - це, по суті, процес поліпшення зовнішнього вигляду і продуктивності ресурсів або сторінок сайту, які забезпечують унікальну цінність для можливих користувачів за рахунок SEO на сторінці, оптимізації конверсії, взаємодії з користувачем, дизайну, редагування контенту і багато чого іншого.

Є 2 основних стратегічних підходи до досягнення успіху в пошуковій оптимізації.

Перший шлях - зосередити кампанію на технічну сторону пошукової оптимізації. Сайти електронної торгівлі, які мають тисячі сторінок, зазвичай процвітають при такому підході.

Другий шлях, - це зосередитися на тому, щоб максимально використовувати те, що вже є на сайті - за допомогою систематизованої повторної оптимізації контенту.

Цей процес фокусується на аудиті і поліпшенні на рівні сторінок, що відрізняє його від того, як виглядають загальні технічні локальні SEO-

аудити (які в основному фокусуються на змінах в масштабах сайту з самого початку).

Він починається з призначення і зв'язування декількох цілей для кожної сторінки, яку оптимізують, наприклад:

1. Забезпечення ранжування однієї сторінки для кількох пошукових запитів (рис. 2.3).

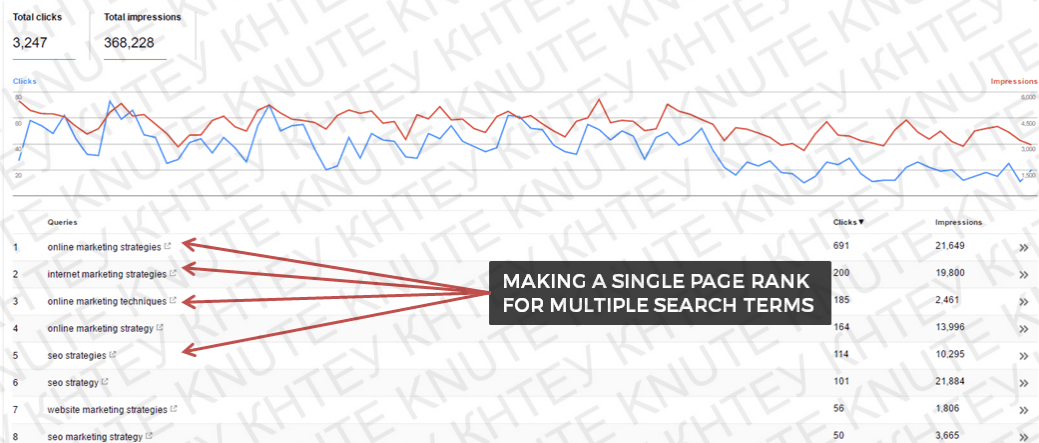


Рис. 2.3. Забезпечення ранжування однієї сторінки для кількох пошукових запитів

2. Зайняти «0-е» місце в семантичному пошуку (рис. 2.4).

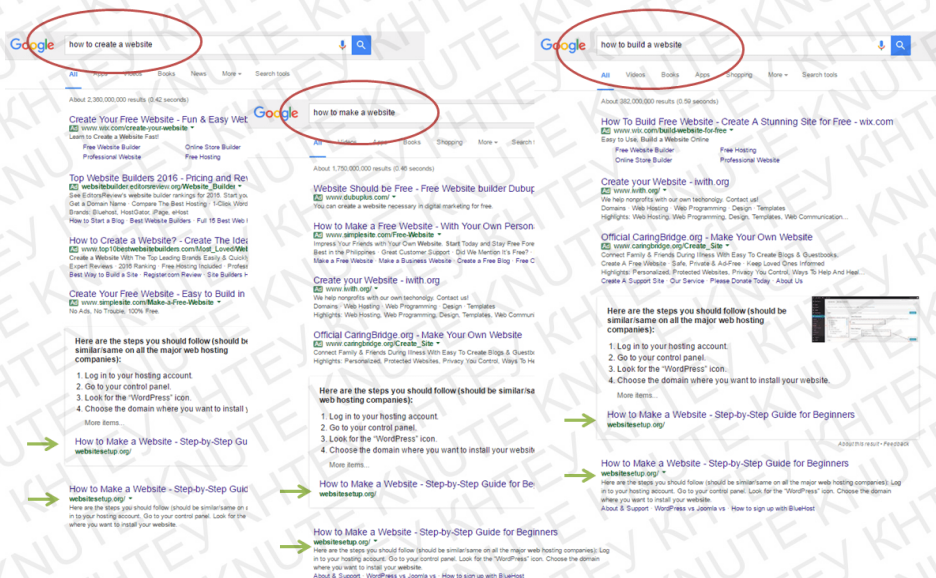


Рис. 2.4. 0-е місце в семантичному пошуку

3. Підвищення ймовірності отримання на сторінках посилань за інформаційними запитами (рисунок 2.5).

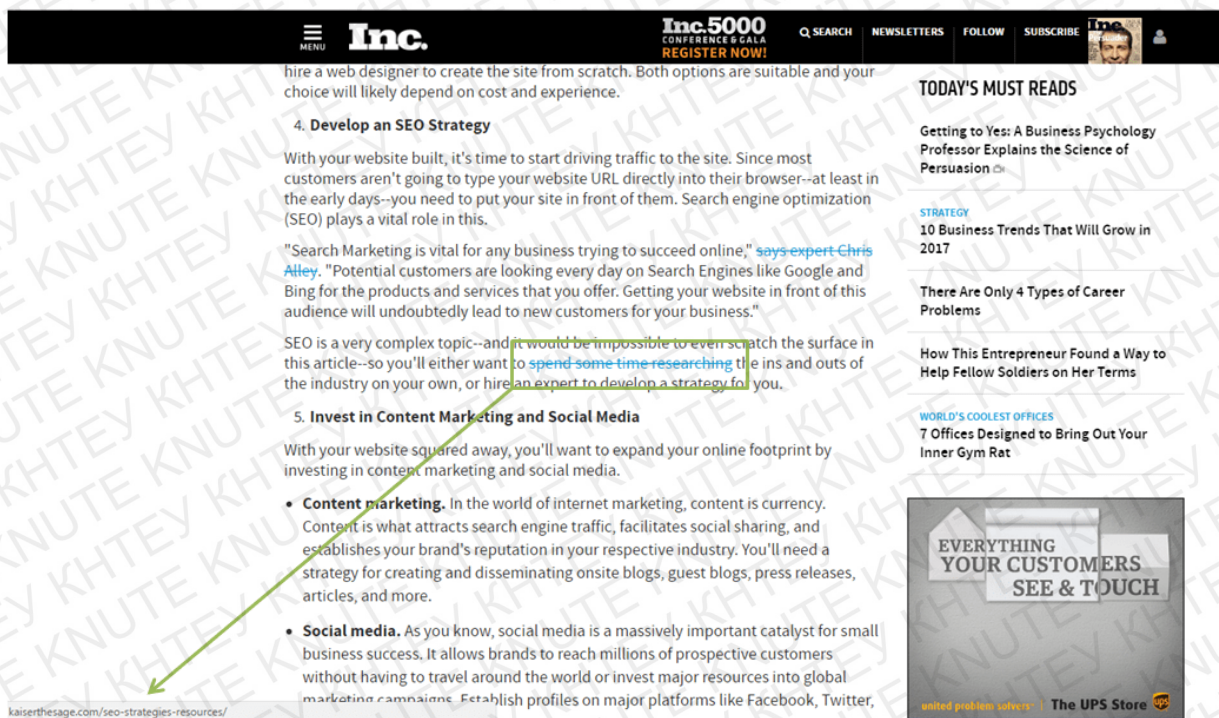


Рис. 2.5. Підвищення ймовірності отримання на сторінках посилань за інформаційними запитами

Одним із сучасних методів оптимізації веб-сайту є використання SILO-архітектури.

Архітектура Silo - це система інформаційної архітектури, яка організовує контент в групи, що відносяться до певної теми в карті сайту веб-сайту. Сторінки розташовані в ієрархії від загального до конкретного (зверху вниз). Навігація є логічною і впорядкованою, тому відвідувачам сайту не потрібно багато часу, щоб навчитися переміщатися по сайту. Веб-сканери розуміють, як індексувати цю форму архітектури веб-сайту [24].

При правильному плануванні і виконанні розрізнена структура веб-сайту може поліпшити підвищити ієрархію контенту веб-сайту. Визначимо основні переваги добре структурованого сховища контенту:

- Більш проста і логічна навігація для відвідувачів сайту;
- Простий пошук потрібного контенту;

- Пошукові системи можуть легко виявляти і сканувати контент сайту;
- Підвищення релевантності ключових слів.

Приклад архітектури SILO представлений на рис. 2.6.

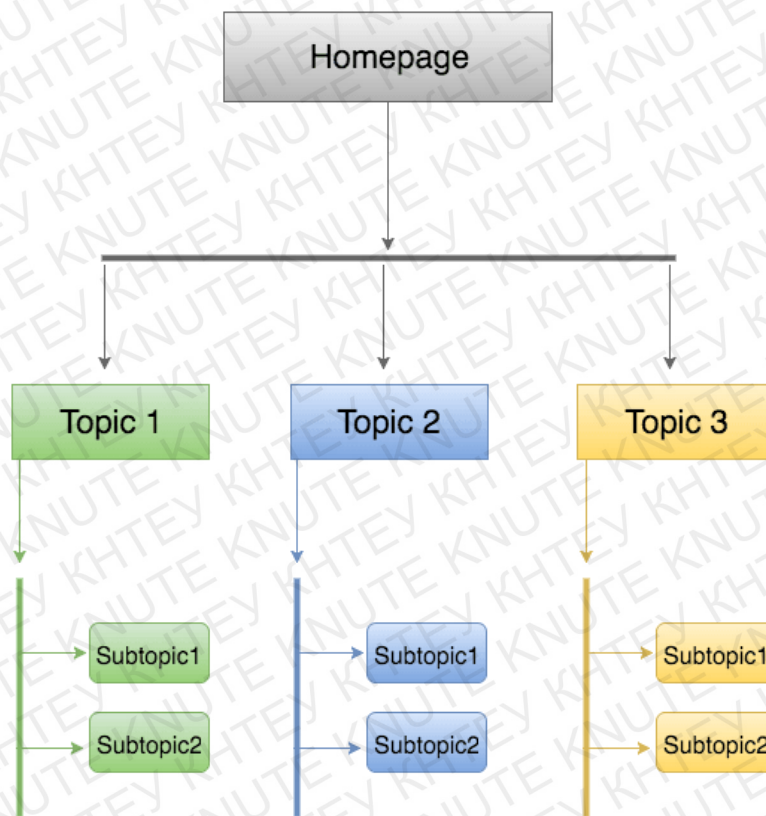


Рис. 2.6. Приклад SILO архітектури

На рис. 2.7 представлена структура сайту, побудованого за допомогою даної архітектури.

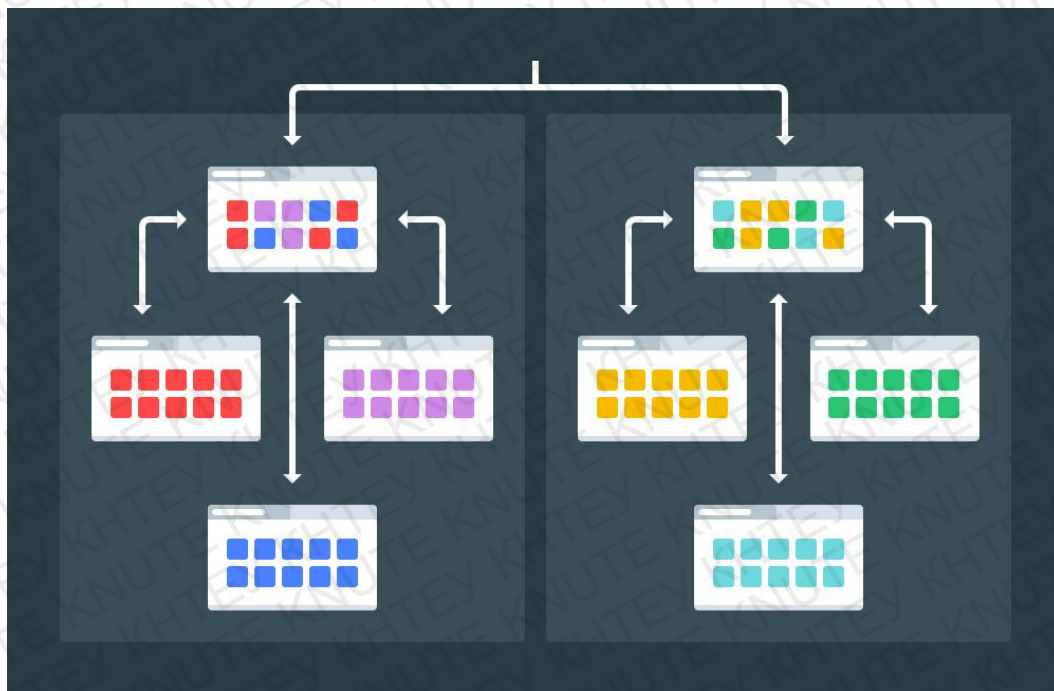


Рис. 2.7. Структура сайту, побудованого за допомогою SILO

При індексуванні сайту пошукові системи враховують релевантність ключового слова по відношенню до змісту на сторінці. Система також бере до уваги контекст веб-сайту. Релевантність ключового слова і контекст повинні бути на місці, щоб збільшити шанси бути знайденими в Інтернеті.

Використання такої архітектури для організації контенту допомагає створювати повідомлення, які мають відношення до загальної теми, а також до категорій. При застосування даної архітектури є більше шансів з'явитися в результаті для ключового слова, коли створюється контекст.

При створенні зворотних посилань на сайт важливо переконатися, що ці зворотні посилання допомагають користувачам знаходити контент на сайті. Розрізнена структура дозволить пов'язувати верхні категорії з повідомленнями в підкатегоріях, які потім можуть посилатися на домашню сторінку. Такий взаємозв'язок корисний, бо дає змогу сканувати весь сайт. Після сканування сайту збільшується ймовірність того, що релевантні статті та повідомлення ранжуватимуться за певними умовами пошуку.

При правильному зв'язуванні збільшиться ймовірність того, що пошукові системи практично не беруть до уваги контент сайту. Пошукові системи використовують посилання для пошуку більшої кількості контенту на сайті [25].

Кращі практики щодо зв'язування припускають, що пов'язується контент тільки в ізольованому просторі. Це гарантує, що посилання ведуть до релевантного контенту. Якщо потрібно посилатися на інші сховища, рекомендується посилатися на категорії верхнього рівня.

Існує два різних методи ізольованого зберігання: фізичні ізольовані каталоги і віртуальні ізольовані.

Обидва дозволяють власникам сайтів створювати вузькі теми за допомогою стратегій зв'язування; однак вони підходять до цієї концепції зовсім інакше. Віртуальне роз'єднання необхідно; фізична роз'єднаність підсилює теми, але не завжди здійсненна. У будь-якому випадку, якщо можливо, перед створенням веб-сайту слід спроектувати розрізненість. Це дає більше контролю над процесом проектування.

Завдяки правильному поділу, власники сайтів можуть підвищити рейтинг своїх сторінок за тематичними ключовими словами, а також із загальних ключових слів сайту. Розрізнені сайти можуть стати домінуючими у видачі по важливим ключовим фразам.

Віртуальна роз'єднаність набагато більш актуальна для того, як пошукова система індексує контент. Структура внутрішніх посилань з'єднує групи пов'язаних сторінок, розділяючи незв'язані сторінки і зміцнюючи цільову сторінку. Ця структура змушує ботів підключати сторінки за певними ключовими словами, підвищуючи ймовірність того, що сторінки будуть ранжуватися за цими ключовими словами. Крім того, в такому стилі організації легко орієнтуватися, оскільки сторінки знаходяться там, де читачі чекають їх побачити.

Отже, віртуальні розрізнені сховища використовують структуру перехресних посилань з деталізацією для створення окремих тематичних категорій. Іншими словами, тема верхньої цільової сторінки підтримується відповідними сторінками з посиланнями на неї. Віртуальне роз'єднання може бути корисно для встановленого сайту, у якого не налаштована файлова система каталогів або який не може порушити встановлені місця розташування сторінок для зміни архітектури сайту.

У віртуальному сховищі кожна допоміжна сторінка пов'язана з цільовою сторінкою категорії, а також пов'язана з іншими допоміжними сторінками для цієї теми. Тема створюється і посилюється за рахунок перехресних посилань на сторінках. При віртуальному поділі сторінки не обов'язково повинні знаходитися в одному каталозі, щоб перебувати в одному сховищі; тема створюється виключно за допомогою посилань.

Перехресна роз'єднаність і внутрішня роз'єднаність, які в сукупності називаються внутрішніми посиланнями на сайти, є однією з важливих частин кращих практик віртуального роз'єднання. Три основних типи посилань, які впливають на високий рейтинг ключових слів, це вхідні посилання, витікаючі посилання і внутрішні посилання на сайти.

Пошукові системи переглядають веб-сторінки по посиланнях. Створення посилань з інших сайтів - відмінний спосіб підвищити рейтинг в пошукових системах. Однак також важливо розмістити посилання на власному веб-сайті.

Коли встановлюється внутрішнє посилання, пошукові системи можуть легше сканувати сторінки сайту. Коли за посиланням переходять з зовнішнього сайту, підвищується ймовірність того, що боти продовжать сканувати сторінки.

Створення або перепроєктування веб-сайту з нуля може бути складним завданням, але хороша новина полягає в тому, що це дає переваги для побудови структури сайту з урахуванням SEO.

Важливо думати про свою ієрархію як про спосіб організації всієї інформації, яку хочеться відображати на своєму сайті. Оскільки це також буде впливати на навігацію сайту і структуру URL-адреси, правильне розуміння цього аспекту життєво важливо. Також важливо, що ця методологія змусить задуматися про те, який контент хочеться створювати в середньостроковій і довгостроковій перспективі, а також в найближчому майбутньому.

Розглянемо деякі рекомендації по організації правильної архітектури [26]:

1. Рекомендується прийняти надзвичайно логічний підхід. Останнє, що хочеться зробити - це надмірно ускладнити свою ієрархію. Необхідно прагнути до простоти і переконатися, що кожне прийняте рішення спрямоване на забезпечення безпроблемної взаємодії як для відвідувачів, так і для пошукових роботів.

2. Необхідно прагнути об'єднати від двох до семи категорій контенту. Занадто багато категорій дуже швидко може здатися переважним, тому, якщо є більше семи категорій, можливо, варто переглянути свій організаційний план. Природно, категорії будуть відповідати продукту, але якщо це занадто розпливчасто, необхідно спробувати поглянути на це під іншим кутом.

3. Необхідно створити баланс в підкатегоріях кожної категорії. Прагнення до балансу ніколи не буває поганим, тому, якщо в одній з категорій є дві підкатегорії, а в іншій - більш ніж в два рази, сайт може швидко виглядати і відчувати себе незбалансованим.

Структура навігації сайту, природно, буде слідувати встановленій ієрархії. Важливо стежити за тим, щоб кожна сторінка, особливо найважливіші, ніколи не була занадто глибоко вкладена в сайт. Найефективніші веб-сайти - це ті, кожна сторінка яких доступна за три або менше кліків.

При наявності добре налаштованої ієрархії наступним завданням буде створення додаткової структури URL-адреси, багатой ключовими словами. Це допоможе забезпечити більш високі позиції в рейтингу для окремих сторінок і допоможе користувачам з легкістю переміщатися по сайту.

При створенні архітектури URL слід враховувати наступні рекомендації:

- Ретельно інтегрувати релевантні ключові слова;
- Створити перспективну структуру;
- Уникати зайвих слів і символів;
- Звернути увагу на чутливість до регістру.

2.3 Рекомендації щільності ключових слів для кращого ранжування

Щільність ключового слова означає кількість повторень ключової фрази на сторінці в процентах від загальної кількості слів на сторінці [27]. Вчені і дослідники до нинішнього часу сперечаються, яка найкраща щільність ключових слів для пошукових систем?

Вони погодилися в наступному:

- Не існує єдиної найкращої щільності ключових слів для високого рангу в Google або Bing;
- Оптимальна щільність ключових слів відрізняється від сторінки до сторінки, від фрази до фрази;
- Необхідно писати природно і включати ключову фразу один або два рази на сторінці;
- Необхідно уникати повторення ключових фраз в текстовому змісті;

- Зосередитися на створенні високоякісного привабливого контенту.

Не існує універсального оптимального відсотка «щільності ключових слів», який будь-хто будь-коли демонстрував і мав пряме позитивне поліпшення рейтингу на публічному форумі.

Хоча ключем до успіху в багатьох нішах часто є просте розміщення ключових слів, в наші дні пошукові системи не так легко «обдурити».

Природно думати, що важливо визначити золоту середину для досягнення більш високих позицій в Google. Питання в тому, яка оптимальна щільність ключових слів.

Якщо дослідити це питання, можна знайти безліч суперечливих думок про ідеальну процентну щільність ключових слів.

Існує формула для визначення локальної щільності ключових слів будь-якої сторінки:

$$Dens = \frac{N_{kr}}{(T_{kn} - (N_{kr}(N_{wp} - 1)))} \cdot 100\%, \quad (2.1)$$

де Dens - щільність ключового слова;

N_{kr} - скільки разів повторювали певну ключову фразу;

N_{wp} - кількість слів у ключовій фразі;

T_{kn} - загальна кількість слів в аналізованому тексті.

Корисно знати значення щільності ключових слів для конкретної фрази на сторінці, щоб уникнути переповнення тексту ключовими словами.

Цілком можливо, що Google не використовує значення % щільності ключових слів навіть для ідентифікації спаму або для застосування фільтрів наповнення ключовими словами. Проте, оцінювачів якості просять перевірити текст на предмет наповнення ключовими словами.

Незважаючи на те, що багато інструментів SEO вказують на те, що не існує «кращої» щільності ключових слів, треба писати природно і

включати ключову фразу один або два рази, щоб зайняти місце в Google і уникнути зниження. Якщо виявиться, що повторюються ключові фрази, то, ймовірно, наповнюється текст ключовими словами [28].

В деяких статтях вказано, що якщо ми будемо використовувати ключове слово багато разів, ми отримаємо більш високий рейтинг в пошуковій видачі. Насправді це може бути погано, оскільки Google може розглядати сторінку (і ваш сайт) як «наповнення ключовими словами» і призведе до штрафу за надмірну оптимізацію.

Важливо пам'ятати наступне:

- Статті з низькою щільністю ключових слів не принесуть бажаних результатів в пошукових системах;
- Статті з дуже високою щільністю ключових слів можуть перешкодити всьому веб-сайту розсилати спам за ключовими словами.

В деяких дослідженнях написано, що оптимально 1-3% щільності ключових слів разом з «семантичними ключовими словами» і ключовими словами LSI.

Google пропонує писати природні статті. Ідеального відсотка не існує, але розміщення ключового слова в природних місцях буде працювати найкраще.

Потрібно підтримувати хороший відсоток щільності ключових слів, щоб вони виглядали природно.

Існує безліч онлайн-калькуляторів щільності ключових слів, які можна використовувати для розрахунку щільності ключових слів. Рекомендується підтримувати щільність близько 1,5%.

Ще одним важливим питанням є місце розміщення ключового слова. Рекомендується його розміщувати в:

- постійному посиланні;
- тезі H1;
- тезі H2;

- мета тези заголовка;
- на початку статті;
- кінці статті.

Рекомендується додати релевантні зображення і використовувати цільове ключове слово в якості якірного тексту.

Треба уникати прийому «набивати свою статтю ключовими словами тільки для збільшення щільності ключових слів». Замість цього рекомендується розмістити ключове слово природним чином, якщо це здається необхідним.

Також треба використовувати відео, слайди і інші медіа-інструменти, щоб зробити контент більш інформативним і презентабельним.

Отже, показники наступні:

- Щільність ключових слів корисна, але слід уникати переповнення;
- Статті повинні виглядати природно.
- Можна виділити жирним шрифтом і курсивом ключове слово і інші важливі «семантичні ключові слова»;
- Можна використовувати варіанти ключових слів (наприклад, пошукова оптимізація, SEO, рейтинг в пошукових системах);
- Використовувати ключові слова LSI;
- Рекомендується застосувати плагін WordPress SEO для перевірки щільності ключових слів (або онлайн-інструмент). Можна також використовувати комбінацію Yoast SEO та помічника по написанню SEO, щоб максимально використовувати щільність ключових слів.

Кращі інструменти перевірки щільності ключових слів:

1. Модулі SEO (WordPress) Якщо використовується WordPress, плагін SEO, такий як Yoast SEO, Rank Math, показує щільність ключових фраз в розділі редактора повідомлень.

Він показує, оптимальна чи щільність цільового ключового слова. Він враховує загальну кількість слів на сторінці і пропонує зміни на основі цього.

Вікно налаштування представлено на рис. 2.8.

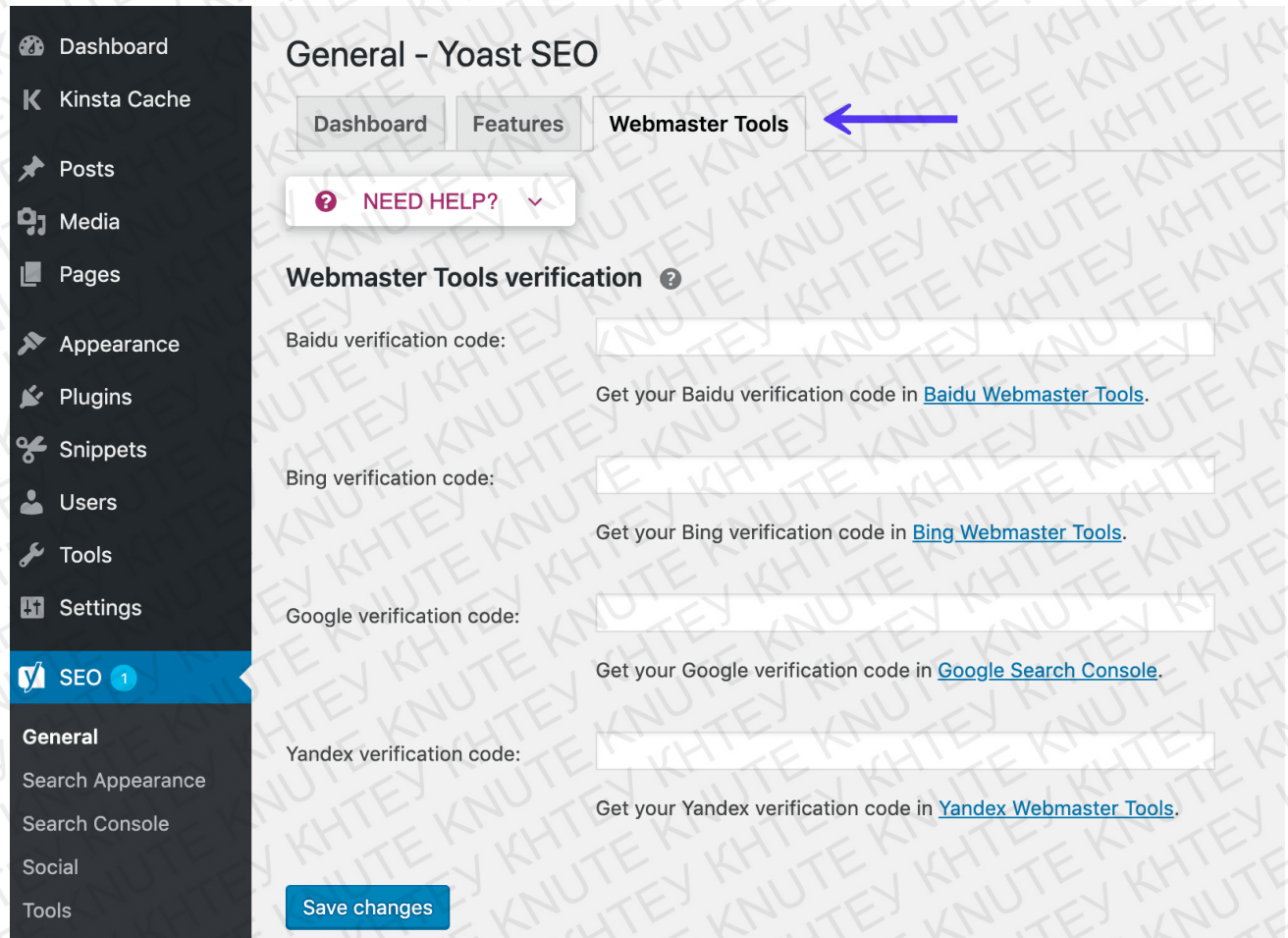


Рис. 2.8. Вікно налаштування Yoast SEO

2. Інструмент Istio. Це ще один інструмент, який допоможе у створенні звіту про щільність ключових слів / фраз [29].

Вікно аналізу представлено на рис. 2.9.

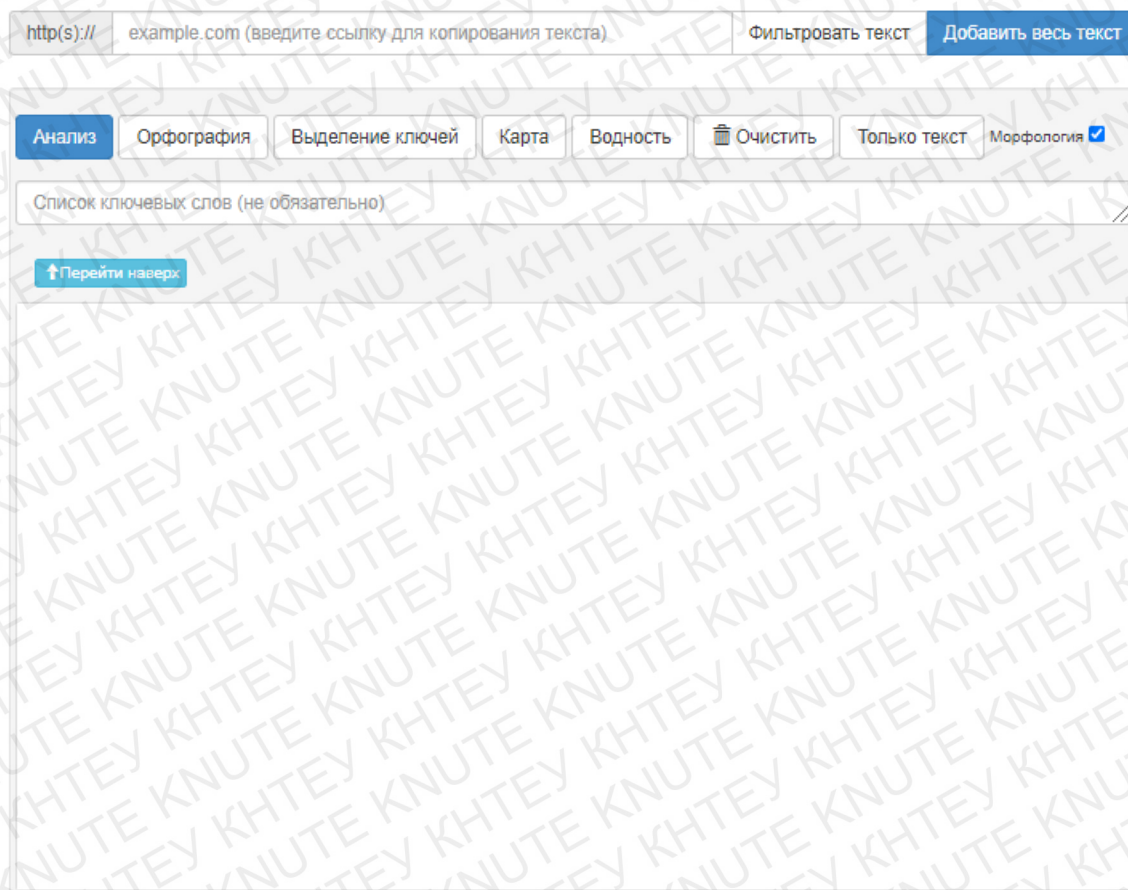


Рис. 2.9. Вікно аналізу Istio

Це кілька з кращих інструментів перевірки щільності ключових слів, з якими можна стикатися. Зрештою, пошукові системи найбільше дбають про інші фактори ранжирування:

- Наскільки корисна стаття;
- Наскільки це інформативно;
- Скільки соціальних сигналів отримують статті;
- Скільки зворотних посилань отримується;

Є сотні сигналів, які Google враховує при ранжуванні сайту. Щільність ключових слів, безумовно, є одним з таких сигналів. Якщо зробити це правильно, це допоможе у просуванні і пошуковій оптимізації.

РОЗДІЛ 3.

РОЗРОБКА СИСТЕМИ ДЛЯ РЕКОМЕНДАЦІЙ ПО НАПОВНЕННЮ САЙТУ

3.1 Розробка моделі рекомендаційної системи

В межах даного розділу розглянемо особливості розробки системи для рекомендацій з наповнення сайту.

Модель рекомендаційної системи будемо розглядати на прикладі сайту, представленого на рис. 3.1 [30].



Dog Products

Dog Services

You are here: [Home](#) / Dog Gifts (Unique & Handpicked Ideas)

Dog Gifts (Unique & Handpicked Ideas)



Рис. 3.1. Сайт barkymate

Розроблена рекомендаційна система може показувати, яка структура web-контенту повинна бути, щоб сайт оптимально був відображений у пошуковій системі (щоб за результатом пошуку, наприклад, ошийника, даний сайт відображався на першій сторінці пошукової системи Google).

Вхід до інформаційної системи представлений на рис. 3.2.

Login

Username

Password

Рис. 3.2. Процедура входу до інформаційної системи

Вигляд вікна введення URL представлений на рис. 3.3.

Project: URL:

Manage Content to Analyze

GWT Analyzer

Country: Device: From: To:

Show: Keyword Filter:

Keyword	InKeyword	InIgnored	Position	Clicks	Impressions	InText	Where
---------	-----------	-----------	----------	--------	-------------	--------	-------

Рис. 3.3. Вигляд вікна введення URL

Після натискання на Retrieve можна побачити весь контент досліджуваного сайту, який розбитий за тегами (рис. 3.4).

Manage Content to Analyze

title

Dog Collars ▶ 10+ Cool & Best Dog Collars of 2018 (High-Quality)

h1

Dog Collars (Handpicked & Reviewed)

p

(2 votes, average: 5.00 out of 5) Loading... The Isle of Dogs, Lassie Come Home and Beethoven have a few things in common. First off, they are movies. Secondly, they revolve around the worst case scenario conceivable to any pet parent: dogs gone missing. Even the sheer idea of a pup going astray can induce palpitations – this is a scientific fact.

Рис. 3.4. Вивід контенту досліджуваного сайту

Також можна змінити період аналізу, пристрій та країну (рис. 3.5).

GWT Analyzer

Country
Device
From
To
Fetch

Show
Show All
Keyword Filter

Keyword
InKeyword
Ignored
Position
Clicks
Impressions
InText
Where

Рис. 3.5. Налаштування аналізу

Результати аналізу представлені на рис. 3.6.

Country 

Device 

From 

To 

Fetch 

5/14/2018

9/3/2020

Show 

Show All 

Keyword Filter 

Keyword	InKeyword	InIgnored	Position	Clicks	Impressions	InText	Where
cool dog collars			34.1	2	434	1	li
awesome dog collars			15.5	1	58	0	
premium dog collars			49.9	1	28	0	
4th of july dog accessories			53.3	0	3	0	
4th of july dog collars			58.9	0	7	0	
active dog collars			35.3	0	3	0	
amazing dog collars			37.8	0	9	0	
american dog collars			98	0	1	0	

Рис. 3.6. Результати аналізу за ключовими словами

Результати розбиття за словами представлені на рис. 3.7.

Content Analyzer				
Divider	3	Fetch	Words count: 1523	☐ Export competitors table
Export	Google Logout			
Filter	Filter	Filter	Filter	Filter
Keyword	InTarget	InText	%	
premium	2	2	0.1	
duty	1	2	0.1	
dog	10	40	2.6	
collar	3	30	2	
heavy	1	3	0.2	
collars	7	26	1.7	
awesome	2	3	0.2	
best	2	11	0.7	
Keyword	InTarget	InText	%	
premium dog	2	2	0.1	
best custom	1	0	0	
dog collars	7	22	1.4	
custom dog	1	0	0	
duty dog	1	1	0.1	
dog collar	3	16	1.1	
cool dog	1	1	0.1	
heavy duty	1	2	0.1	
Keyword	InTarget	InText	%	
premium dog collars	1	0	0	
awesome dog collar	1	2	0.1	
custom dog collars	1	0	0	
best custom dog 1	0	0	0	
premium dog collar	1	2	0.1	
heavy duty dog	1	1	0.1	
best dog collars	1	4	0.3	
cool dog collars	1	1	0.1	

Рис. 3.7. Результати розбиття за словами

У відсотках ми бачимо, яка щільність повторення даного слова. Якщо щільність слова від 1 до 3%, то відсоток відображається зеленим кольором, якщо менше – жовтим кольором.

3.2 Специфіка програмної реалізації за допомогою фреймворку Django та Angular

В межах даного підрозділу будуть наведені особливості побудови програм за допомогою фреймворку Django та Angular.

Django - це програмний каркас з багатими можливостями, що підходить для розробки складних сайтів і веб-додатків, написаний на мові програмування Python [31].

Django - фреймворк для веб-додатків на мові Python. Один з основних принципів фреймворка - DRY (do not repeat yourself). Веб-системи на Django будуються з одного або декількох додатків, які рекомендується робити відчужуваними. Це одна з помітних архітектурних відмінностей цього фреймворка від деяких інших (наприклад, Ruby on Rails). Також, на відміну від багатьох інших фреймворків, обробники URL в Django конфігуруються явно (за допомогою регулярних виразів), а не автоматично задаються зі структури контролерів.

Django проектувався для роботи під управлінням Apache (з модулем `mod_python`) і з використанням PostgreSQL в якості бази даних. В даний час, крім PostgreSQL, Django може працювати з іншими СУБД: MySQL (MariaDB), SQLite, Microsoft SQL Server, DB2, Firebird, SQL Anywhere і Oracle. Для роботи з базою даних Django використовує власний ORM, в якому модель даних описується класами Python, і по ній генерується схема бази даних.

Приклад написання додатку представлений на рис. 3.8.

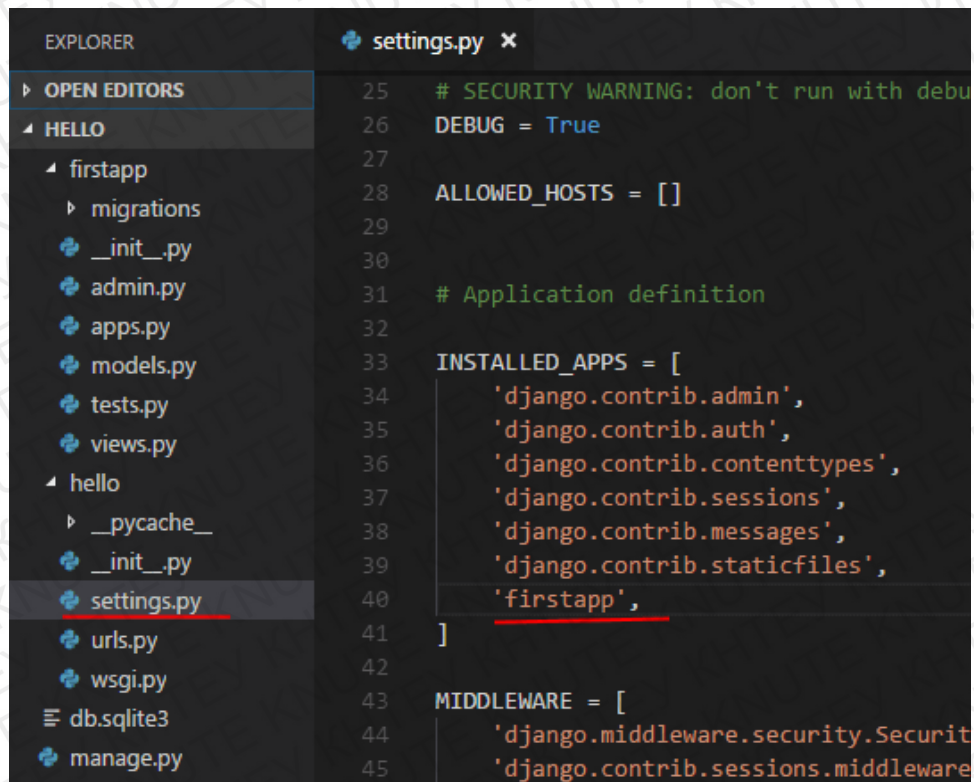


Рис. 3.8. Приклад написання додатку на Django

Архітектура Django схожа на «Модель-Представлення-Контролер» (MVC). Контролер класичної моделі MVC приблизно відповідає рівню, який в Django називається Представлення (View), а презентаційна логіка Уявлення реалізується в Django рівнем Шаблонів (Templates).

Через це рівневу архітектуру Django часто називають «Модель-Шаблон-Подання» (MTV).

Спочатку розробка Django велася для забезпечення більш зручної роботи з новинними ресурсами, що досить сильно відбилося на архітектурі: фреймворк надає ряд засобів, які допомагають у швидкій розробці веб-сайтів інформаційного характеру.

Наприклад, розробнику не потрібно створювати контролери та сторінки для адміністративної частини сайту, в Django є вбудований додаток для керування вмістом, який можна включити в будь-який сайт, зроблений на Django, і який може управляти відразу декількома сайтами на одному сервері. Адміністративний додаток дозволяє створювати,

змінювати і видаляти будь-які об'єкти наповнення сайту, протоколюючи всі дії, і надає інтерфейс для управління користувачами і групами (з пооб'єктним призначенням прав).

Веб-фреймворк Django використовується в таких великих і відомих сайтах, як Instagram, Disqus, Mozilla, The Washington Times, Pinterest, lamoda і ін.

Деякі можливості Django:

- ORM, API доступу до БД з підтримкою транзакцій;
- вбудований інтерфейс адміністратора, з уже наявними перекладами багатьма мовами;
- диспетчер URL на основі регулярних виразів;
- розширювана система шаблонів з тегами і наслідуванням;
- система кешування;
- інтернаціоналізація;
- архітектура додатків, які можна встановлювати на будь-які

Django-сайти;

- «Generic views» - шаблони функцій контролерів;
- авторизація та аутентифікація, підключення зовнішніх модулів аутентифікації: LDAP, OpenID та інші;
- система фільтрів («middleware») для побудови додаткових обробників запитів, як наприклад включені в дистрибутив фільтри для кешування, стиснення, нормалізації URL і підтримки анонімних сесій;
- бібліотека для роботи з формами (наслідування, побудова форм по існуючій моделі БД);
- вбудована автоматична документація по тегам шаблонів і моделям даних, доступна через адміністративний додаток.

Деякі компоненти фреймворка між собою пов'язані слабо, тому їх можна досить просто замінювати на аналогічні. Але з деякими (наприклад,

з ORM) це зробити не дуже просто. Крім можливостей, вбудованих в ядро фреймворка, існують пакети, що розширюють його можливості.

На базі Django розроблено досить багато готових рішень, які розповсюджуються під вільною ліцензією, серед яких системи для управління інтернет-магазинами, універсальні системи управління вмістом, а також більш цілеспрямовані проекти.

Angular - один з найпопулярніших інструментів розробки програмного забезпечення, що є частиною системи JavaScript. Він був представлений Google в 2009 році. Згідно з опитуванням, проведеним StackOverflow в 2019 році, 30,7 відсотка розробників програмного забезпечення застосовують AngularJS і нову версію Angular 2+ для створення користувацьких інтерфейсів [32].

Фреймворк Angular спрощує розробку веб-додатків. Поєднання впровадження залежностей, декларативних шаблонів, наскрізних інструментів і інтегрованих передових практик вирішує майже всі проблеми при створенні веб-додатку.

Фреймворк Angular включає версії з 2 по 8. Восьме оновлення було випущено в травні 2019 року.

Angular 2: випуск Angular 2 вніс численні зміни в початковий фреймворк, оскільки він був переписаний на TypeScript. Архітектурний стиль перейшов на компонентний.

Angular 4: Angular CLI 1.0.0 був представлений в четвертій версії як основний елемент проекту Angular. З випуском Angular Universal додатки Angular могли відображатися поза браузером.

Angular 5-6: випуск п'ятої і шостої версії був зосереджений на оптимізації Angular CLI і роботи компілятора.

Angular 7: з Angular 7 інтерфейс командного рядка було розширено підказками, які надавали підказки в інтерфейсі командного рядка для пояснення функцій і призначення елементів. Отже, використання CLI

стало більш інтуїтивним. Додатки отримали різні поліпшення продуктивності і розміру кодової бази.

Angular 8: остання версія Angular: в Angular 8 були представлені два елементи рендеринга Ivy, Bazel (інтерфейс збірки). Ще одне важливе поліпшення - це диференційне завантаження, яке використовується для завантаження пакетів для конкретного браузера, щоб підтримувати застарілі браузери і швидше завантажувати контент.

Кожна технологія має певні переваги і недоліки. Розглянемо ці переваги і недоліки Angular.

Плюси Angular:

- Реалізація архітектури MVC;
- Архітектура модель-уявлення-контролер не тільки надає значення структурі при створенні клієнтської програми, але також встановлює основу для інших функцій, таких як прив'язка даних і області;
- З архітектурою MVC можна ізолювати логіку додатку від шару призначеного для користувача інтерфейсу і підтримки поділу проблем. Контролер отримує всі запити для програми та працює з моделлю для підготовки будь-яких даних, необхідних для подання. Подання використовує дані, підготовлені контролером, і відображає остаточну презентабельну відповідь;
- Покращена архітектура дизайну. Деякі з великих веб-додатків містять безліч компонентів. Angular спрощує управління цими компонентами, навіть якщо новий програміст приєднується до проекту після того, як процес розробки вже почався. Архітектура побудована таким чином, що допомагає програмісту легко знаходити і розробляти код;
- Модулі. Модуль - це механізм, який групує пов'язані директиви, компоненти, канали і служби таким чином, щоб їх можна було комбінувати з іншими модулями для створення програми.

Є кілька способів додати різні елементи в модуль. Angular вирішує проблему використання глобальних функцій, обмежуючи область дії всіх функцій модулем, де вони були визначені і використані;

- Впровадження служб і залежностей (DI). Службі або компоненту іноді можуть знадобитися інші залежні служби для виконання завдання. Dependency Injection дизайн шаблон використовується для виконання цих залежностей. Він розподіляє завдання між різними службами.

Клієнтська служба не буде створювати залежний об'єкт, скоріше він буде створений і введений інжектором Angular. Інжектор Angular відповідає за створення примірників сервісів і їх впровадження в такі класи, як компоненти і сервіси;

- Призначені для користувача директиви. Призначені для користувача директиви покращують функціональність HTML і підходять для динамічних клієнтських додатків. Всі вони починаються з префікса ng, щоб HTML міг їх ідентифікувати.

- Angular написаний з використанням TypeScript, який є надмножиною JavaScript. Він повністю відповідає JavaScript, а також допомагає виявляти і усувати типові помилки при кодуванні. У той час як невеликі проекти JavaScript не вимагають такого удосконалення, для додатків корпоративного масштабу необхідно, щоб розробники зробили свій код чистішим і частіше перевіряли якість.

Недоліки Angular:

- Обмежені можливості SEO. Головний недолік використання Angular - обмежені можливості SEO і погана доступність для пошукових робіт.

- Angular «багатослівний» і складний. Часті скарги, які можна побачити від розробників Angular, - це багатослівність інструменту. І ця проблема не сильно змінилася з часів AngularJS.

- Великий час на навчання. Якщо треба залучити нових розробників, знайомих з JavaScript, для використання нового Angular, їм буде складно в порівнянні з адаптацією на React або Vue. Це пов'язано з тим, що коло тем і аспектів, які необхідно охопити, досить велике.
- У документації CLI відсутні подробиці. Деякі розробники висловлюють заклопотаність поточним станом документації CLI. Хоча командний рядок - дуже корисний для розробників Angular, не знайдеться достатньо інформації в їх офіційній документації на GitHub, і доведеться витратити більше часу на вивчення потоків на GitHub, щоб отримати відповіді.

Таким чином, хоча у платформи є свої недоліки, Angular - це повнофункціональний і динамічний фреймворк. А зручність використання, гнучкість і ремонтпридатність роблять його унікальним і дають можливість створювати відмінні і успішні веб-додатки.

3.3 Технологія використання рекомендаційної системи по наповненню сайту

Результати аналізу за всіма ключовими словами представлені на рис.

3.8.

Keyword	InTarget	InText	%	Where
cool dog collars	1	1	0.1	li
awesome dog collars	1	0	0	
premium dog collars	1	0	0	
awesome dog collar	1	2	0.1	p, li
best custom dog collars	1	0	0	
best dog collars	1	4	1.3	title, h2, p, p
dog collar	3	16	1.1	a, a, a, h2, a, a, p, p, p
dog collars	7	22	1.4	li, title, a, h1, h2, li, li, li, h2, li

Рис. 3.8. Результати аналізу за всіма ключовими словами

Також система дозволяє проаналізувати структуру контенту і для сайту-конкурента (рис. 3.9).

Filter	Keyword	InTarget	InText	%
	premium	2	0	0
	duty	1	0	0
	dog	10	0	0
	collar	3	1	0,4
	heavy	1	0	0
	collars	7	1	0,4
	awesome	2	0	0
	best	2	0	0

Filter	Keyword	InTarget	InText	%
	premium dog	2	0	0
	best custom	1	0	0
	dog collars	7	0	0
	custom dog	1	0	0
	duty dog	1	0	0
	dog collar	3	0	0
	cool dog	1	0	0
	heavy duty	1	0	0

Filter	Keyword	InTarget	InText	%
	premium dog collars	1	0	0
	awesome dog collar	1	0	0
	custom dog collars	1	0	0
	best custom dog 1	0	0	0
	premium dog collar	1	0	0
	heavy duty dog	1	0	0
	best dog collars	1	0	0
	cool dog collars	1	0	0

Рис. 3.9. Результати аналізу для сайту-конкурента

Розроблена система дозволяє експортувати результати аналізу в Google Excel таблиці (рис. 3.10).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Keywords	Position	Clicks	Impressions	InText	Where	IsTarget	IsIgnored		Keyword	InTargetCount	InTextCount	Percent	Where
2	cool dog collars	34.08	2	434	1	1	TRUE	FALSE		cool dog collars	1	0	0	
3	awesome dog co	15.47	1	58	0		TRUE	FALSE		awesome dog co	1	0	0	
4	premium dog co	49.89	1	28	0		TRUE	FALSE		premium dog co	1	0	0	
5	4th of july dog a	53.33	0	3	0		FALSE	FALSE		awesome dog co	1	0	0	
6	4th of july dog a	58.86	0	7	0		FALSE	FALSE		best custom dog	1	0	0	
7	active dog collar	35.33	0	3	0		FALSE	FALSE		best dog collars	1	0	0	
8	amazing dog col	37.78	0	9	0		FALSE	FALSE		dog collar	3	0	0	
9	american dog cc	98	0	1	0		FALSE	FALSE		dog collars	7	0	0	
10	awesome dog at	31.89	0	66	0		FALSE	FALSE		heavy duty dog	1	0	0	
11	awesome dog co	14.35	0	17	2	p, 1	TRUE	FALSE		premium dog co	1	0	0	
12	best custom dog	86.83	0	6	0		TRUE	FALSE						
13	best dog access	66.4	0	10	0		FALSE	FALSE						
14	best dog collar	94.8	0	5	0		FALSE	FALSE						
15	best dog collar b	93	0	1	0		FALSE	FALSE						
16	best dog collars	70.97	0	36	4	little, h2, p, p	TRUE	FALSE						
17	best dog collars	11.7	0	23	0		FALSE	FALSE						
18	best dog ever cc	27	0	1	0		FALSE	FALSE						
19	best dog id tags	79.5	0	2	0		FALSE	FALSE						
20	best heavy duty	77	0	1	0		FALSE	FALSE						
21	best indestructib	93	0	1	0		FALSE	FALSE						
22	best martingale	88.27	0	11	0		FALSE	FALSE						
23	best material for	82.33	0	3	0		FALSE	FALSE						
24	best personalize	100	0	1	0		FALSE	FALSE						
25	best personalize	95	0	2	0		FALSE	FALSE						

Рис. 3.10. Результати експорту отриманих результатів аналізу

Також є можливість зберегти отримані результати (Save) і присвоїти назву кожному з проєктів.

ВИСНОВКИ

У випускному кваліфікаційному проєкті представлено результати теоретичних і прикладних досліджень, що полягають у розробці рекомендаційної web-системи інформаційного наповнення сайту для оптимального відображення в пошукових системах. В результаті проведених досліджень були отримані такі **висновки**:

1. Кожне джерело веб-трафіку є індикатором працездатності сайту. Великий обсяг реферального трафіку означає, що про веб-сайт часто говорять сторонні веб-сайти або сайти соціальних мереж. Хороша стратегія - додати ці метрики джерел трафіку на панель інструментів SEO, щоб глибше поринути в свої зусилля.

2. Веб-сайти, що знаходяться на вершині, отримують 32% трафіку; другий отримує 21%, а третій - тільки 8%. якщо веб-сайт стане видимим нагорі, це буде означати більшу автентичність і довіру, які є очевидними елементами для більшої кількості потенційних клієнтів і трафіку.

3. Щоб зробити сайт зручним для відвідувачів, а також поліпшити його позицію в результатах пошуку, його потрібно оптимізувати. Пошукова оптимізація допомагає правильно інтерпретувати контент і представляти його в пошуку.

4. Добре структуровані веб-сайти, як правило, будуть мати низький показник відмов, тривалий час перебування (час перебування відвідувачів на сайті), хороший рейтинг кліків (CTR) і будуть ресурсом, з яким аудиторія активно хоче проводити час, взаємодіючи. Інтуїтивно зрозумілий інтерфейс - невід'ємна частина будь-якого хорошого веб-сайту, і все це починається з правильної структури, архітектури сайту.

5. Оптимально 1-3% щільності ключових слів разом з «семантичними ключовими словами» і ключовими словами LSI.

6. Розроблена рекомендаційна система дозволяє встановити щільність ключових слів, дозволяє проаналізувати структуру контенту веб-сайту свого та конкурента, а також у зручній формі виконати аналіз контенту сайту загалом.

В результаті написання роботи були виконані наступні задачі:

1. Виконаний аналіз механізмів залучення нових клієнтів за допомогою пошукових web-систем;
2. Здійснено огляд існуючих технік організації інформаційного наповнення сайту, що відображаються на ранжуванні у пошукових системах;
3. Виконано розробку системи для рекомендацій по наповненню сайту.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Pageviews vs Unique Pageviews in Google Analytics: веб-сайт. URL: <https://insights.quiet.ly/blog/pageviews-unique-pageviews-google-analytics/> (дата звернення: 04.10.2020).
2. Бабаев А. Раскрутка: секреты эффективного продвижения сайтов / А. Бабаев, М. Боде, Е. Костин. – СПб.: Питер, 2013. – 272 с.
3. Севостьянов И. Поисковая оптимизация. Практическое руководство по продвижению сайта в Интернете / И. Севостьянов. – СПб.: Питер, 2016. – 272 с.
4. Social Traffic and Conversions: веб-сайт. URL: <https://www.klipfolio.com/resources/kpi-examples/social-media/traffic-conversions/> (дата звернення: 04.10.2020).
5. Experiment Shows Up To 60% Of “Direct” Traffic Is Actually Organic Search: веб-сайт. URL: <https://searchengineland.com/60-direct-traffic-actually-seo-195415> (дата звернення: 04.10.2020).
6. Голополосов, Д.М. 80 способов повысить конверсию сайта; Питер - М., 2013. - 160 с.
7. Зуев М. Б., Маурус П. А., Прокофьев А. Г. Продвижение сайтов в поисковых системах. Спасательный круг для малого бизнеса; Бином. Лаборатория знаний - М., 2011. - 299 с.
8. MOBILE SEO:The Definitive Guide: веб-сайт. URL: <https://backlinko.com/mobile-seo-guide> (дата звернення: 05.10.2020).
9. Салех Х., Шукайри А.Повышение конверсии веб-сайта; Манн, Иванов и Фербер - М., 2012. - 320 с.
10. Факторы ранжирования в поисковых системах: веб-сайт. URL: <https://seoquick.com.ua/search-engine-rankings/> (дата звернення: 05.10.2020).

11. How Important is a Top Listing in Google? : веб-сайт. URL: <https://www.impactbnd.com/blog/important-top-listing-google> (дата звернення: 05.10.2020).

12. No. 1 Position in Google Gets 33% of Search Traffic : веб-сайт. URL: <https://www.searchenginewatch.com/2013/06/20/no-1-position-in-google-gets-33-of-search-traffic-study/> (дата звернення: 05.10.2020).

13. Введение в поисковую оптимизацию: веб-сайт. URL: <https://support.google.com/webmasters/answer/7451184?hl=ru#hierarchy> (дата звернення: 12.10.2020).

14. Организация контента на веб-ресурсе: плоская иерархия vs. глубокая иерархия: веб-сайт. URL: <https://lpgenerator.ru/blog/2015/09/03/> (дата звернення: 12.10.2020).

15. How to Create a Site Structure That Will Enhance SEO: веб-сайт. URL: <https://neilpatel.com/blog/site-structure-enhance-seo/> (дата звернення: 12.10.2020).

16. What Is Website Architecture? A 3-Minute Rundown: веб-сайт. URL: <https://blog.hubspot.com/marketing/website-architecture> (дата звернення: 12.10.2020).

17. Тероу Шэри Видимость в Интернете. Поисковая оптимизация сайтов; Символ-Плюс - , 2009. - 288 с.

18. R. Dutta, G. N. Rouskas, I. Baldine, A. Bragg, and D. Stevenson, "The silo architecture for services integration, control, and optimization for the future internet," in Proceedings of IEEE ICC, Glasgow, Scotland, June, 2007.

19. Кристофер Б. Джонс 140 технологий раскрутки сайтов; Рид Групп - Москва, 2011. - 352 с.

20. Фридман В. А., Александров А. В., Сергеев Г. Г., Костин С. П. Строительство Web-сайтов (+ CD-ROM); Триумф - Москва, 2011. - 288 с.

21. 9 Content Optimization Strategies to increase visibility: веб-сайт. URL: <https://www.outbrain.com/help/advertisers/content-optimization/> (дата звернення: 12.10.2020).

22. Мерсер, Дэвид Drupal 6. Создание надежных и полнофункциональных веб-сайтов, блогов, форумов, порталов и сайтов-сообществ; М.: Вильямс - Москва, 2009. - 272 с.

23. The complete guide to content optimization: веб-сайт. URL: <https://kaiserthesage.com/content-optimization-tips/> (дата звернення: 12.10.2020).

24. B. Ahlgren et al. Invariants – a new design methodology for network architectures. In FDNA-04, pages 65–70, Portland, OR, 2004.

25. SEO Siloing: How to build a website silo architecture : веб-сайт. URL: <https://www.bruceclay.com/seo/silo/> (дата звернення: 12.10.2020).

26. Silo Architecture for SEO and User Experience : веб-сайт. URL: <https://slickplan.com/sitemap/website-structure/silo-architecture> (дата звернення: 12.10.2020).

27. What Is The Best Keyword Density For Search Engines? : веб-сайт. URL: <https://www.hobo-web.co.uk/keyword-density-seo-myth/> (дата звернення: 12.10.2020).

28. Китинг, Джоди Flash MX. Искусство создания web-сайтов; ТИД ДС - Москва, 2012. - 848 с.

29. Плотность ключевых слов: формула подсчета и сервисы: веб-сайт. URL: <https://mcs.bitrix24.ua/company/personal/bizproc/> (дата звернення: 12.10.2020).

30. Dog Gifts : веб-сайт. URL <https://barkymate.com/dog-gifts/> (дата звернення: 12.10.2020).

31. Django — фреймворк для веб-разработки на Python: веб-сайт. URL: <https://web-creator.ru/articles/django> (дата звернення: 12.10.2020).

32. What are the Advantages and Disadvantages of Angular? : веб-сайт.

URL: <https://www.edureka.co/blog/advantages-and-disadvantages-of-angular/>

(дата звернення: 12.10.2020).

ДОДАТОК

Програмний код реалізації Web-додатку

```
import json

import httplib2

from googleapiclient.errors import HttpError
from googleapiclient.http import BatchHttpRequest, HttpRequest
from oauth2client.service_account import ServiceAccountCredentials

class IndexingService:

    SCOPES = ['https://www.googleapis.com/auth/indexing']
    KEY_FILE_LOCATION = 'api_key.json'
    ENDPOINT = "https://indexing.googleapis.com/v3/urlNotifications:publish"
    ENDPOINT_BATCH = "https://indexing.googleapis.com/batch"

    def __init__(self):
        credentials =
        ServiceAccountCredentials.from_json_keyfile_name(IndexingService.KEY_FI
        LE_LOCATION,
                                                    scopes=IndexingService.SCOPES)

        self.http = credentials.authorize(httplib2.Http())

    def execute_request(self, links):
        responses = []

        def batch_callback(request_id, response, exception):
            if response:
```

```

        responses.append(response)
    elif exception:
        responses.append({
            'url': links[int(request_id) - 1],
            'status': exception.resp['status'],
            'message': str(exception)
        })

def generate_response(resp, content):
    try:
        response_content = json.loads(content.decode('utf-8'))
        result = {
            'url': response_content['urlNotificationMetadata']['url'],
            'status': resp['status'],
            'message': 'OK'
        }
    except Exception as e:
        resp.reason = "Something went wrong while processing successful
response. Original response %s." \
                    "Exception: %s " % (resp.reason, str(e))
        raise HttpError(resp.reason, content)
    return result

batch =
BatchHttpRequest(batch_uri=IndexingService.ENDPOINT_BATCH)
for link in links:
    link_object = {
        'url': link,
        'type': 'URL_UPDATED'
    }

```

```

    }
    batch.add(request=HttpRequest(self.http, generate_response,
IndexingService.ENDPOINT, method='POST',
                                body=json.dumps(link_object)),
callback=batch_callback)

```

```

    batch.execute(http=self.http)
    return responses

```

```

import os.path

```

```

from whoosh.analysis import StandardAnalyzer
from whoosh.index import create_in, open_dir
from whoosh.fields import Schema, STORED, TEXT
from whoosh.qparser import QueryParser

```

```

# print([(t.text, t.stopped) for t in myanalyzer(u"casino game! Online ttt'www-
ss..!. paypal \ casino's game", removestops=True)])

```

```

class InTextCounterService:

```

```

    def __init__(self):
        self.analyzer = StandardAnalyzer(expression=r'([.,!?:;]+|\w+((\w+
|\w+)\w+)*)', minsize=1, stoplist=[])
        self.schema = Schema(tag=STORED,
content=TEXT(analyzer=self.analyzer))
        self.index = self.create_index()

```

```

self.parser = QueryParser('content', self.index.schema)
self.searcher = None

def open_searcher(self):
    self.searcher = self.index.searcher()

def close_searcher(self):
    self.searcher.close()

def create_index(self):
    if not os.path.exists("index"):
        os.mkdir("index")
    create_in("index", self.schema)
    return open_dir("index")

def add_documents(self, documents):
    writer = self.index.writer()
    for document in documents:
        writer.add_document(tag=document['tag'], content=document['text'])
    writer.commit()

def process_keyword(self, keyword):
    query = self.parser.parse("%s" % keyword)
    results = self.searcher.search(query, limit=None, terms=True)

    inside_tags = []
    in_text_count = len(results)
    for result in results:
        inside_tags.append(result.get('tag'))

```

```
return ', '.join(inside_tags), in_text_count
```

```
def fill_keywords_count(keywords, content):
```

```
    service = InTextCounterService()
```

```
    service.add_documents(content)
```

```
    service.open_searcher()
```

```
    for keyword in keywords:
```

```
        result = service.process_keyword(keyword['keyword'])
```

```
        keyword['where'] = result[0]
```

```
        keyword['inText'] = result[1]
```

```
    service.close_searcher()
```

```
def count_words(content):
```

```
    result = 0
```

```
    analyzer = StandardAnalyzer()
```

```
    for unit in content:
```

```
        # print ([t.text) for t in analyzer(unit['text'], removestops=True)])
```

```
        result += len([t for t in analyzer(unit['text'])])
```

```
    return result
```

```
from django.contrib.auth.models import Group
```

```
from django.core.exceptions import ValidationError
```

```
from django.db import models
```

```

class Project(models.Model):
    name = models.CharField(max_length=255, unique=True)
    default = models.BooleanField()
    groups_have_access = models.ManyToManyField(
        Group,
        blank=True
    )

    def clean(self):
        default_project = Project.objects.filter(default=True)
        if self.default and default_project and default_project.get() != self:
            raise ValidationError('Default project is already set. Unset it first.')

    def __str__(self):
        return self.name

class SavedPage(models.Model):
    url = models.URLField()
    related_project = models.ForeignKey(Project, on_delete=models.CASCADE)

class ContentUnit(models.Model):
    related_page = models.ForeignKey(SavedPage,
    on_delete=models.CASCADE)
    tag = models.CharField(max_length=30)
    text = models.TextField()

```

```
class TargetKeyword(models.Model):
    related_page = models.ForeignKey(SavedPage,
    on_delete=models.CASCADE)
    keyword = models.CharField(max_length=255)
```

```
class IgnoredKeyword(models.Model):
    related_page = models.ForeignKey(SavedPage,
    on_delete=models.CASCADE)
    keyword = models.CharField(max_length=255)
```

```
import os.path
import re
```

```
from whoosh.analysis import StandardAnalyzer
from whoosh.index import create_in, open_dir
from whoosh.fields import Schema, STORED, TEXT
from whoosh.qparser import QueryParser
```

```
class PhraseCounterService:
```

```
    def __init__(self):
        self.analyzer = StandardAnalyzer(expression=r'\w+', minsize=1,
        stoplist=[])
        self.schema = Schema(phrase=TEXT(analyzer=self.analyzer))
        self.index = self.create_index()
        self.parser = QueryParser('phrase', self.index.schema)
        self.searcher = None
```

```

def open_searcher(self):
    self.searcher = self.index.searcher()

def close_searcher(self):
    self.searcher.close()

def create_index(self):
    if not os.path.exists("index"):
        os.mkdir("index")
    create_in("index", self.schema)
    return open_dir("index")

def add_documents(self, documents):
    writer = self.index.writer()
    for document in documents:
        writer.add_document(phrase=document['keyword'])
    writer.commit()

def keyword_count(self, keyword):
    query = self.parser.parse("%s" % keyword)
    results = self.searcher.search(query)
    return len(results)

def form_same_count_keywords(keywords, words_count):
    result = set()
    for keyword in keywords:
        result.update(divide_phrase(keyword.get('keyword', ''), words_count))

```

```

return result

def divide_phrase(phrase, divide_count):
    result_list = []
    words_list = re.compile('\s+').split(phrase.strip())
    for index in range(0, len(words_list)):
        if (divide_count <= len(words_list) - index):
            included_phrase = words_list[index:index+divide_count]
            result_list.append(' '.join(included_phrase))
        else:
            return result_list

    return result_list

import requests
import re
from bs4 import BeautifulSoup, Tag, NavigableString, Comment

class TextData:
    def __init__(self, text, tag):
        self.text = text
        self.tag = tag

    def __str__(self):
        return '<{0}>{1}</{0}>'.format(self.tag, self.text)

class WebPageParser:

```

```
#Sites do not allow to retrieve html content without this header
user_agent_header = {'User-Agent': 'Mozilla/5.0 (Macintosh; Intel Mac OS X
10_13_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/72.0.3626.121
Safari/537.36'}
```

```
skipped_tags = ['script', 'style', 'noscript', 'iframe', 'ym-measure', 'nav', 'footer',
'img', 'table', 'button']
```

```
skipped_classes = ['top-casino-cards', 'top-sites-wrap']
```

```
def __init__(self, url):
```

```
    self.url = url
```

```
    self.result = []
```

```
def parse_web_page_text(self):
```

```
    title, body = self.retrieve_title_and_body()
```

```
    self.result.append(TextData(title.get_text().strip(), title.name))
```

```
    for element in body.contents:
```

```
        self.parse_element_content(element)
```

```
    return self.result
```

```
def parse_element_content(self, element):
```

```
    if isinstance(element, Tag) and element.name not in
```

```
WebPageParser.skipped_tags and not self.contains_skipped_class(element):
```

```
        if self.tag_contains_text(element):
```

```
            self.result.append(TextData(element.get_text().strip(), element.name))
```

```
        else:
```

```
            for child_element in element.contents:
```

```
                self.parse_element_content(child_element)
```

```

def contains_skipped_class(self, element):
    if element.attrs.get('class'):
        for class_name in element.attrs.get('class'):
            if class_name in WebPageParser.skipped_classes:
                return True
    return False

def tag_contains_text(self, tag):
    for element in tag.contents:
        if type(element) is NavigableString and not re.match(r'^\s+$', element):
            return True
    return False

def retrieve_title_and_body(self):
    soup = BeautifulSoup(self.retrieve_page_html(), 'html.parser')
    return soup.find('title'), soup.find('body')

def retrieve_page_html(self):
    response = requests.get(self.url, headers =
WebPageParser.user_agent_header)
    if response.status_code != 200:
        raise Exception('Could not retrieve "%s" page html. Status code is %s' %
(self.url, response.status_code))
    return response.text

from apiclient.discovery import build
from oauth2client.service_account import ServiceAccountCredentials

class WebmasterService:

```

```
SCOPES = ['https://www.googleapis.com/auth/webmasters.readonly',  
'https://www.googleapis.com/auth/webmasters']  
KEY_FILE_LOCATION = 'api_key.json'  
  
def __init__(self):  
    credentials =  
ServiceAccountCredentials.from_json_keyfile_name(WebmasterService.KEY_  
FILE_LOCATION, scopes=WebmasterService.SCOPES)  
    self.webmasters_service = build('webmasters', 'v3', credentials=credentials)  
  
def execute_request(self, url, request):  
    return self.webmasters_service.searchanalytics().query(siteUrl=url,  
body=request).execute()
```