

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ
ТАРАСА ШЕВЧЕНКА**

Факультет комп'ютерних наук та кібернетики
Кафедра інтелектуальних програмних систем

Випускна кваліфікаційна робота
на здобуття освітнього рівня бакалавра
за спеціальністю 121 Інженерія Програмного Забезпечення
на тему:

**РОЗРОБКА СИСТЕМИ ДЛЯ ВІДСТЕЖЕННЯ АКТИВНОСТІ
КОРИСТУВАЧІВ ВЕБ-САЙТІВ**

Виконав студент 4-го курсу
Віктор ІЛЬВОВСЬКИЙ

(підпис)

Науковий керівник:
к. т. н., доцент кафедри інтелектуальних програмних систем
Євген ДЕМКІВСЬКИЙ

(підпис)

Засвідчую, що в цій роботі немає
запозичень з праць інших авторів без
відповідних посилань.

Студент

(підпис)

Роботу розглянуто й допущено до захисту
на засіданні кафедри інтелектуальних
програмних систем

« ____ » _____ 202_ р.
протокол № ____

Завідувач кафедри
Олександр ПРОВОТАР

(підпис)

РЕФЕРАТ

Обсяг роботи 48 сторінок, 27 джерел посилань, 26 ілюстрацій, 2 таблиці, 3 додатки. Ключові слова: ВЕБ-АНАЛІТИКА, ВЕБ-ВІДСТЕЖЕННЯ, ПОЗНАЧЕННЯ СТОРІНОК, RDNS, ГЕОЛОКАЦІЯ, GOOGLE ANALYTICS, ADOBE ANALYTICS, CLICKTALE.

Об'єктом роботи є система для ведення статистики відвідувань веб-сайтів.

Метою роботи є дослідження поняття веб-аналітики і методів відстеження активності користувачів веб-сайтів, огляд існуючих засобів веб-аналітики та розробка системи для аналітики відвідуваності веб-сайтів.

Результат роботи: розглянуто поняття відстеження відвідувань веб-сайтів і показано актуальність даної проблеми. Проведено порівняння найбільш популярних сервісів для веб-аналітики, таких як: Google Analytics, Adobe Analytics і Click Tale. Реалізовано систему для відстеження відвідувань веб-сайтів і процес її розгортання.

ЗМІСТ

| | |
|--|----|
| ВСТУП | 5 |
| РОЗДІЛ 1: ВЕБ-АНАЛІТИКА..... | 7 |
| 1.1 Веб-аналітика..... | 7 |
| 1.2 Поведінкова аналітика..... | 9 |
| 1.3 А/Б тестування..... | 10 |
| РОЗДІЛ 2: ВЕБ-ВІДСТЕЖЕННЯ | 11 |
| 2.1 Веб-відстеження | 11 |
| 2.2 Веб-відстежувачі | 13 |
| 2.2.1 Файли cookie | 13 |
| 2.2.2 Файли super cookie | 13 |
| 2.2.3 Вбудовані скрипти | 13 |
| 2.2.4 Fingerprinting скрипти..... | 14 |
| 2.2.5 Веб-маяки..... | 14 |
| 2.3 Захист конфіденційності | 14 |
| РОЗДІЛ 3: СЕРВІСИ ДЛЯ ВЕБ-АНАЛІТИКИ..... | 16 |
| 3.1 Сервіси для веб-аналітики..... | 16 |
| 3.2 Сервіс Google Analytics | 16 |
| 3.3 Сервіс Adobe Analytics..... | 18 |
| 3.4 Сервіс Clicktale | 19 |
| 3.5 Порівняння..... | 20 |
| РОЗДІЛ 4: СИСТЕМА ВЕДЕННЯ СТАТИСТИКИ ВІДВІДУВАНЬ ВЕБ-САЙТІВ..... | 22 |
| 4.1 Система «Website Visit Statistics»..... | 22 |
| 4.2 Огляд архітектури | 23 |
| 4.2.1 Діаграма системи | 23 |
| 4.2.2 Геолокація..... | 24 |
| 4.3 Інтерфейс користувача | 25 |
| 4.3.1 Автентифікація..... | 25 |
| 4.3.2 Управління веб-сайтами..... | 26 |
| 4.3.3 Часові статистики..... | 27 |
| 4.3.4 Статистики програмного агента | 28 |
| 4.3.5 Статистики країн | 30 |
| 4.3.6 Статистики сторінок | 31 |
| 4.3.7 Реферальні веб-сайти..... | 32 |
| 4.3.8 А/Б тестування | 32 |
| 4.3.9 Загальна інформація | 33 |

| | |
|--|----|
| 4.4 Деталі реалізації | 34 |
| 4.4.1 Бекенд..... | 34 |
| 4.4.2 Інтеграція | 36 |
| 4.4.3 Фронтенд..... | 36 |
| 4.4.4 Інфраструктура..... | 37 |
| 4.4.5 Неперервна інтеграція | 40 |
| 4.4.6 Розгортання нової версії системи..... | 40 |
| 4.4.7 Локальний запуск системи..... | 41 |
| ВИСНОВКИ..... | 43 |
| ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ..... | 44 |
| ДОДАТОК А..... | 46 |
| Приклад інтеграції з системою через HTML код..... | 46 |
| ДОДАТОК Б | 47 |
| Програмний скрипт <i>visit.js</i> для передачі даних про відвідування..... | 47 |
| ДОДАТОК В..... | 48 |
| Конфігураційний файл <i>config.yml</i> для налаштування неперервної інтеграції | 48 |

ВСТУП

Актуальність. Кожного разу, коли відвідувач запитує певний елемент веб-сторінки, це називається «попаданням» і фіксується в файлі журналу. Воно може включати текст на веб-сторінці, зображення, аудіо- чи відео-файл. Однак спочатку Інтернет складався з переважно статичних сторінок, обмежених текстом та гіперпосиланнями. Тому, коли відвідувач потрапляв на сторінку, система припускала, що він взаємодіє з усім вмістом сторінки. По мірі збільшення використання Інтернету власників веб-сайтів ці показники стали цікавити все більше. Аналіз файлів журналів відкрив шлях, який привів до початку комерційної веб-аналітики. Стартовою точкою можна вважати дату заснування компанії WebTrends у 1993 р. [1].

Коли веб-сторінки почали включати візуальні та інші елементи, крім тексту, стало ясно, що кількість звернень, накопичених сервером, більше не відображає кількість запитуваних сторінок. Позначення сторінок (page tagging) стало новим методом збору даних для точного звітування про різноманітний веб-трафік та тенденції. Це все ще є найпоширенішим методом збору даних на сьогодні.

Веб-аналітика стала більш усталеною як важливий інструмент для веб-оптимізації, надаючи дедалі складніші сервіси, що маніпулюють величезними обсягами даних. Зараз на веб-дизайн та дизайн додатків сильно впливає аналітика: навряд чи будуть створюватися публічні веб-сайти чи програми без А/Б тестування [2]. Тепер, коли розробка UX (досвід користувача) та UI (інтерфейс користувача) базується на результатах веб-аналітики, цей зв'язок стає все міцнішим. Майбутнє стосується оптимізації цих відносин, а також оптимізації накопиченої інформації, щоб забезпечити максимальну задоволеність як користувачем, так і бізнесом. Машинне навчання та когнітивні обчислення є центром сучасного розвитку веб-аналітики, і постачальники аналітики продовжують впроваджувати інновації в цій галузі.

Метою роботи є опис технологій для відстеження відвідувань веб-сайтів, порівняння сучасних сервісів для веб-аналітики, а також розробка власної системи для аналізу відвідувань веб-сайтів.

Об'єктом роботи є система для ведення статистики відвідувань веб-сайтів. Для досягнення визначеної цілі необхідно виконати наступні кроки:

1. Дослідити різноманіття технік для веб-відстеження.
2. Вибрати найбільш популярні сервіси для веб-аналітики і провести їх порівняння.
3. Побудувати власну систему для відстеження відвідувань веб-сайтів, використовуючи відомі технології та сервіси.

РОЗДІЛ 1: ВЕБ-АНАЛІТИКА

1.1 Веб-аналітика

Веб-аналітика – це вимірювання та аналіз даних для розуміння поведінки користувачів на веб-сторінках [3]. Платформи аналітики вимірюють активність та поведінку на веб-сайті, наприклад: скільки користувачів відвідують веб-сайт, як довго вони залишаються на ньому, скільки сторінок вони відвідують, які сторінки відвідують та переходять вони за посиланнями чи ні. Компанії використовують платформи веб-аналітики для вимірювання та порівняння ефективності роботи сайту та вивчення ключових показників ефективності, що направляють їхній бізнес, наприклад – коефіцієнт конверсії покупок.

З іншого боку веб-аналітика – це збір веб-даних та звітування про них з метою розуміння та оптимізації використання Інтернету. Однак веб-аналітика – це не просто процес вимірювання веб-трафіку, вона може бути використана як інструмент для дослідження бізнесу та ринку, а також для оцінки та підвищення ефективності веб-сайту. Веб-аналітика надає інформацію про кількість відвідувачів веб-сайту та кількість переглядів сторінок [4]. Це допомагає оцінити тенденції руху та популярності, що корисно для дослідження ринку. Більшість методів веб-аналітики включають чотири основні етапи:

- Збір даних – це збір основних, елементарних даних. Зазвичай, це якісь кількісні дані.
- Обробка даних – цей етап, як правило, рахує співвідношення. Мета цього етапу – перетворити необроблені дані на інформацію, розрахувати певні метрики.
- Розробка КПЕ (ключові показники ефективності) – цей етап зосереджується на використанні співвідношень або підрахунків та введенні їх у ділові стратегії. Часто ключові показники ефективності мають справу з аспектами конверсії, але не завжди. Це залежить від організації.

- Формування онлайн стратегії – цей етап стосується цілей, завдань та онлайн стандартів для організації чи бізнесу. Ці стратегії, як правило, пов'язані із зароблянням грошей, їх економією або збільшенням частки ринку.

Існує два основних технічних способи збору даних. Перший, традиційний метод – це аналіз файлів журналу сервера. Виконується зчитування журналів, в які веб-сервер записує запити на файли браузером. Другий метод, позначення сторінок (page tagging), використовує JavaScript, вбудований у веб-сторінку, для надсилання запитів на сторонній виділений аналітичний сервер, щоразу, коли веб-сторінка відображається веб-браузером або, за бажанням, коли відбувається клацання мишею. Обидва методи збирають дані, які можуть бути оброблені для створення звітів про веб-трафік.

Основною метою веб-аналітики є збір та аналіз даних, що стосуються веб-трафіку та моделей використання. Самі ж дані в основному надходять із чотирьох джерел:

- Дані запиту HTTP – безпосередньо надходять із повідомлень запиту HTTP (заголовки запитів).
- Дані, пов'язані із запитами HTTP на рівні мережі та на сервері: не є частиною запиту HTTP, але вони потрібні для успішної передачі запитів – наприклад, IP-адреса запитувача.
- Дані рівня додатка, що надсилаються із запитами HTTP: генеруються та обробляються програмами прикладного рівня (такими як JavaScript, PHP та ASP.Net), включаючи сеанси та переходи. Вони зазвичай фіксуються внутрішніми журналами, а не публічними службами веб-аналітики.
- Зовнішні дані: можуть допомогти розширити дані та інтерпретувати використання веб-сайту. Наприклад, IP-адреси, як правило, асоціюються з географічними регіонами та постачальниками послуг Інтернету. Також розглядаються частота відкриття листів електронної

пошти, рейтинг переходів, дані про прямі поштові кампанії, історія продажів та інші типи даних.

1.2 Поведінкова аналітика

Поведінкова аналітика – це нещодавнє досягнення в галузі бізнес-аналітики, яке відкриває нові уявлення про поведінку споживачів на платформах електронної комерції, онлайн-іграх, веб- і мобільних додатках та в Інтернеті Речей. Швидке збільшення обсягу необроблених даних про події, сформовані цифровим світом, дає можливість методам, які виходять за рамки типового аналізу за демографічними та іншими традиційними показниками. Вони повідомляють нам, які люди вживали які дії в минулому. Аналіз поведінки фокусується на розумінні того, як і чому діють споживачі, даючи точні прогнози щодо того, як вони можуть діяти в майбутньому. Це дозволяє маркетологам робити правильні пропозиції для потрібних сегментів споживачів у потрібний час.

Поведінкова аналітика може бути корисною для автентифікації та з міркувань безпеки. Вона використовує неідентифікований, але індивідуально унікальний фактор, щоб визначити, хто є користувачем. Ідентифікація користувача виконується у фоновому режимі з використанням таких факторів, як рух миші чи швидкість набору тексту, звички, історія входу в мережу, IP-адреса, використовуваний браузер тощо.

Поведінкова аналітика використовує величезні обсяги необроблених даних подій користувачів, отриманих під час сеансів, коли користувачі використовують додаток, гру або веб-сайт, включаючи дані про трафік, такі як шлях навігації, кліки, взаємодія в соціальних мережах, рішення про покупку та реагування на маркетинг [5]. Крім того, дані про події можуть включати показники реклами, наприклад час переходу від кліку до конверсії, а також порівняння між іншими показниками, такими як грошова вартість замовлення та кількість часу, проведеного на сайті. Потім ці дані аналізуються, дивлячись на перебіг сеансу з моменту, коли користувач вперше увійшов на платформу, до

здійснення покупки, або на те, які інші продукти користувач купував або дивився до цієї покупки. Поведінковий аналіз дозволяє передбачити майбутні дії та тенденції на основі збору таких даних.

1.3 А/Б тестування

Однією з важливих технік, що використовують аналітики для оптимізації веб-сайтів, є експеримент. А/Б тестування – це контрольований експеримент з двома варіантами в онлайн-налаштуваннях. А/Б тест – це скорочення до простого контрольованого експерименту. Як впливає з назви, порівнюються дві версії (А і Б) однієї змінної, які є ідентичними, за винятком однієї деталі, яка може вплинути на поведінку користувача. А/Б тести вважаються найпростішою формою контрольованого експерименту. Однак, додаючи більше варіантів до тесту, цей метод стає більш складним.

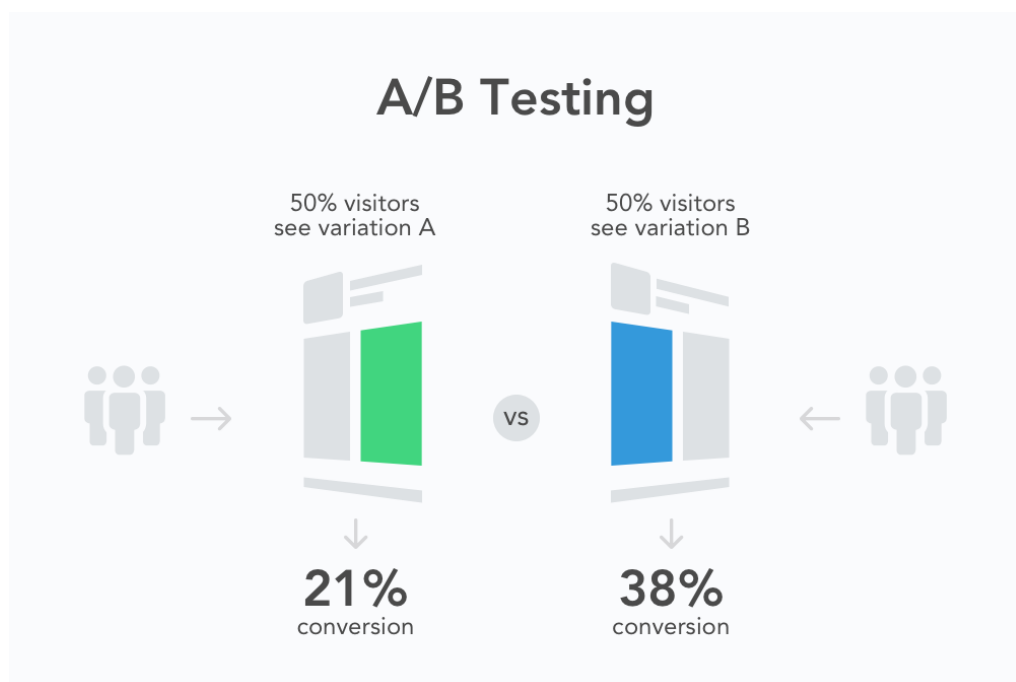


Рис. 1 – Принцип А/Б тестування

А/Б тести є корисними для розуміння залучення користувачів та задоволення онлайн функціями (рис. 1). Великі соціальні мережі, такі як LinkedIn, Facebook та Instagram, використовують А/Б тестування, щоб зробити користувацький досвід кращим [2].

РОЗДІЛ 2: ВЕБ-ВІДСТЕЖЕННЯ

2.1 Веб-відстеження

Веб-аналітика більше фокусується на роботі веб-сайту в цілому. Тоді, як веб-відстеження дає змогу зрозуміти, як використовується веб-сайт, і побачити, скільки часу користувач провів на певній сторінці [6]. За допомогою цього можна дізнатись, кого найбільше цікавить вміст веб-сайту. Веб-відстеження – це практика, за допомогою якої веб-сайти визначають та збирають інформацію про користувачів. Дані, як правило, подаються у формі деякої підмножини історії веб-перегляду.

Здебільшого, відстежувач (tracker) – це програмний скрипт на веб-сайтах, призначений для отримання даних про уподобання користувача та про те, ким він є під час взаємодії з сайтом [7]. Іноді ці скрипти цілеспрямовано розміщуються веб-сайтом, на якому перебуває користувач, а інколи скрипт може бути з веб-сайту, який він ніколи не відвідував. Розуміння способів збору даних може допомогти обмежити вплив відстежувачів в майбутньому.

Щоразу, коли хтось користується Інтернетом, він залишає запис про веб-сайти, які він відвідує, разом із всім, що він натискає [8]. Для відстеження цієї інформації багато веб-сайтів зберігають невеликий фрагмент даних, вбудовують невидимі об'єкти або використовують облікові записи користувачів та налаштування обладнання. Нижче наведені деякі типи веб-відстеження:

- Рекламні компанії активно збирають інформацію про користувачів та створюють профілі, які використовуються для індивідуалізації реклами. Діяльність користувачів включає відвідувані веб-сайти, переглянуті відео, взаємодії в соціальній мережі та онлайн транзакції. Веб-сайти, такі як Netflix, збирають інформацію про те, що показують користувачам, і це допомагає пропонувати більше фільмів чи серіалів, які можуть їм сподобатись. Така пошукова система, як Google, веде облік того, що користувачі шукають, що може допомогти їм запропонувати більш релевантні результати в майбутньому.

- Правоохоронні органи можуть використовувати веб-відстеження для шпигунства за особами та для розкриття злочинів.
- Веб-аналітика більше фокусується на роботі веб-сайту в цілому. Веб-відстеження дає змогу зрозуміти, як використовується веб-сайт, і побачити, скільки часу користувач провів на певній сторінці. За допомогою цього можна дізнатись, кого може найбільше зацікавив вміст веб-сайту.
- Тести для перевірки придатності до використання – це практика перевірки того, наскільки простим є дизайн. За користувачами спостерігають, в процесі виконання ними деяких завдань. Це допомагає виявляти проблеми використання, щоб їх можна було виправити.

Існують багато способів відстеження веб-відвідувань. Проте найбільш загальними є наступні два:

- IP-адреси – це те, що ідентифікує користувачів в Інтернеті. Веб-сайти можуть визначати географічне розташування користувача за IP-адресою. Кожному пристрою, підключеному до Інтернету, присвоюється власна IP-адреса, що дозволяє пристроям обмінюватися даними між собою. IP-адреса є основним компонентом роботи Інтернету, і завдяки унікальності IP-адрес їх можна використовувати для відстеження користувачів.
- HTTP cookies – це інформація, яку зберігає веб-браузер. Коли користувач відвідує веб-сайт, сайт може зберігати файли cookie, щоб він міг в майбутньому розпізнати пристрій користувача. Коли користувач повертається на сайт, останній може прочитати файл cookie, щоб згадати користувача з останнього відвідування. Файли cookie можуть використовуватися для налаштування користувацького перегляду веб-сторінок і для показу цільових оголошень. Далі наведено деякі види веб-активності, які можуть зберігати файли cookie:
 - Сторінки та вміст, який було переглянуто
 - Те, що користувач шукав в Інтернеті

- Коли було натиснуто на Інтернет-рекламу
- Коли користувач відвідував сайт
- Основні файли cookie створюються доменом, який відвідує користувач. Ці cookies вважаються хорошими. Вони допомагають забезпечити кращий досвід для користувачів.

2.2 Веб-відстежувачі

2.2.1 Файли cookie

Це загальноживаний метод відстеження під час перегляду веб-сторінок. Файли cookie – це невеликі шматочки даних, які веб-сайти зберігають у браузері. Щоразу, коли користувач відвідує веб-сторінку, файли cookie можуть зберігатися на його пристрої. Але не всі файли cookie створюються рівними. Веб-сайт, який відвідується, може мати власні файли cookie, які пропонуються для найбільш релевантного для користувача вмісту, вони допомагають запам'ятати входи, запобігти шахрайству тощо.

2.2.2 Файли super cookie

Super cookie – це cookie браузера, яке можна постійно зберігати на комп'ютері. Super cookies важче виявити та позбутися, оскільки вони не можуть бути автоматично видалені так само, як звичайні файли cookie. Вводячи super cookie у пристрій користувача, веб-сайти можуть отримати доступ до його особистої інформації, поведінки та уподобань. Super cookie також можуть відстежувати час, коли користувач є найактивнішим в Інтернеті. Ці дані наступного рівня – це золота копалина для рекламодавців, які можуть використовувати цю інформацію для створення цільових оголошень на основі індивідуалізації наповнення для користувачів та їхніх уподобань.

2.2.3 Вбудовані скрипти

Існує безліч ресурсів для відстеження, які можуть бути вбудовані як видимі або невидимі елементи на сторінці. Поширеними прикладами є піксельні

відстежувачі, які представляють собою зображення, вбудовані на веб-сторінку або в електронну пошту; або скрипти відстеження, які запускаються під час відвідування веб-сайтів і можуть зробити багато речей, від запису IP-адреси до збору характеристик пристрою.

2.2.4 Fingerprinting скрипти

Fingerprinting скрипти збирають дані з веб-браузера, зокрема його тип і версію, операційну систему та версію пристрою користувача, роздільну здатність екрана, встановлені шрифти та навіть можуть бачити всі програми, які були завантаженими на даний пристрій. Ці дані збираються у fingerprint (з англ. – відбиток пальця), унікальний для певного пристрою, і можуть використовуватися для відстеження на веб-сайтах.

2.2.5 Веб-маяки

Веб-маяки (web beacons) – це дуже маленькі, як правило, невидимі об’єкти, вбудовані у веб-сторінку чи лист електронної пошти [8]. У найпростішій формі це крихітні чіткі зображення, часто розміром один піксель. Вони завантажуються як зображення, при відкритті веб-сторінки або листа електронної пошти, здійснюючи запит на віддалений сервер за зображенням. Виклик сервера попереджає компанію про те, що їх електронна пошта щойно відкрита або веб-сторінка була відвідана.

2.3 Захист конфіденційності

ЄС (Європейський Союз) та штат Каліфорнія дотримуються суворих правил захисту конфіденційності в цифровій сфері за допомогою Загального регламенту захисту даних (GDPR) та Каліфорнійського закону про конфіденційність споживачів (CCPA) [9]. Метою цих законів є надання людям свободи та контролю над їх власними даними, примушуючи рекламні компанії та великий бізнес бути прозорими та давати уявлення про те, як хтось відстежується, ким і з якою метою, а також надавати можливість запобігти цьому.

GDPR вимагає, щоб веб-сайти отримували попередню згоду користувачів перед тим, як збирати та обробляти їхні персональні дані. ССРА змушує компанії розміщувати на своєму веб-сайті посилання *«Не продавати мою особисту інформацію»*, щоб користувачі могли відмовитись від продажу своїх даних третім особам. Обидва закони також надають користувачам права бути проінформованими та видаляти зібрані дані.

РОЗДІЛ 3: СЕРВІСИ ДЛЯ ВЕБ-АНАЛІТИКИ

3.1 Сервіси для веб-аналітики

Розглядаючи різні інструменти веб-аналітики, які потрібні для бізнесу, можна побачити безліч доступних варіантів, але часто важко зрозуміти, як ними користуватися. Тому, стає важливою роль бізнес-аналітика, що спеціалізується на аналізі даних відповідних звітів [10].

Очевидно, що користувач не буде постійно використовувати всі ці інструменти, тож корисно знати про деякі найкращі варіанти та про те, як вони вписуються у загальну веб-стратегію. Використання кількох інструментів надає подальший рівень розуміння клієнтів та рівня успішності продукту.

Сервіс для веб-аналітики може надати аналіз якісних та кількісних даних з веб-сайту користувача та сайтів конкурентів. Це сприяє постійному вдосконаленню онлайн-досвіду клієнтів та потенційних клієнтів, що перетворюється на бажані результати. Найпопулярнішим сервісом веб-аналітики є Google Analytics [3], хоча на ринку існує безліч інших, що пропонують спеціалізовану інформацію, таку як активність у реальному часі або теплокарти. Нижче наведено перелік найбільш використовуваних сервісів:

- Google Analytics – стандартний сервіс для веб-аналітики, безкоштовний для невеликих компаній;
- Adobe Analytics – аналітична платформа з великою кількістю різних налаштувань;
- Clicktale – використовує теплокарти, відстежує натискання клавіш та рухи миші.

3.2 Сервіс Google Analytics

GA (Google Analytics) – це добре відомий стандартний сервіс для відстеження, аналізу та звітування даних веб-сайту. За даними W3Techs, він контролює 85% ринку [11] і вважається найбільш комплексним доступним рішенням (рис. 2).

Звичайно, в сервісі є безкоштовна версія. З іншого боку, є Google Analytics 360 націлений на великі організації. Він включає SLA (угода про рівень послуг), технічну підтримку та розширені обмеження кількості запитів.

Переваги. Оскільки ми дедалі більше оцифруємося і використовуємо кілька пристроїв на шляху до конверсії, дані аналітики можуть бути не такими надійними, як мали б бути. Найбільшою перевагою GA 360 є його зв'язок з іншими продуктами Google. GA 360 з'єднує різні джерела даних в одному місці, що робить його комплексним ресурсом для всіх точок дотику користувачів. Сервіс також цінується за точну візуалізацію даних, власні звіти та велику бібліотеку експертних книг, статей та блогів.

Недоліки: вибір GA 360 означає, що користувач зобов'язується використовувати Google. Він погоджується передавати всі свої дані про діяльність цифровому гігантові, що може викликати занепокоєння в організаціях, для яких важлива конфіденційність. Ціноутворення не має гнучкості, і його не можна регулювати залежно від функціональних можливостей, які насправді використовуються. Більше того, навіть якщо ліміти даних розширені, вибір преміум версії все одно залишає проблему обробки даних для великого обсягу трафіку.

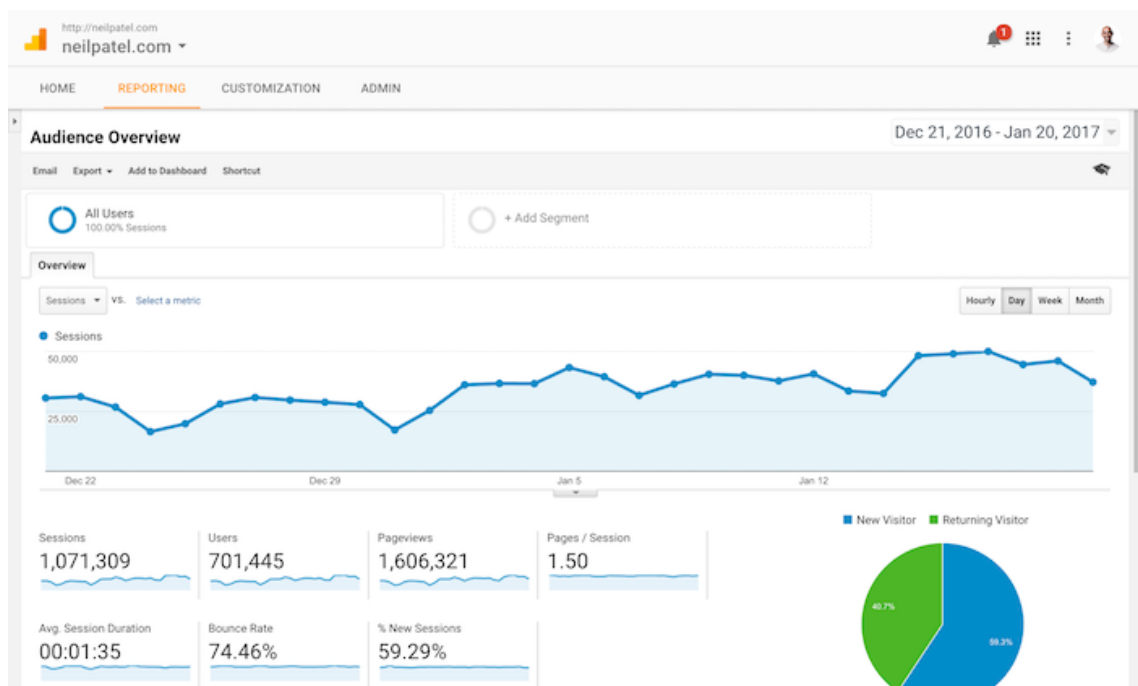


Рис. 2 – Сервіс Google Analytics

3.3 Сервіс Adobe Analytics

Даний сервіс дозволяє вимірювати ефективність роботи веб-сайтів та маркетингової діяльності, а також має безліч налаштувань для користувачів з досвідом. Adobe Analytics можна інтегрувати з іншими маркетинговими продуктами Adobe. Він пропонує вдосконалену сегментацію, автоматизацію в режимі реального часу, спеціальний аналіз та інші функції. Adobe Analytics використовується в корпоративному середовищі та надає підтримку та консультації своїм клієнтам (рис. 3).

Переваги: для досвідченого маркетолога це чудовий вибір з різними функціями та функціональними можливостями. Особливо, коли мова заходить про можливість сегментації, сервіс пропонує гнучкість та багато налаштувань. Adobe Analytics є сильним конкурентом завдяки своїм корпоративним функціям (наприклад, спільному використанню звітів та сегментів з автоматичним оновленням) та звітам з повною інформацією (unsampled reports).

Недоліки: Adobe Analytics вважається одним з найдорожчих сервісів на ринку. Він вимагає кваліфікованих аналітиків (у тому числі технічно) для використання всіх функцій та правильного налаштування платформи. Він добре інтегрується з Adobe Marketing Cloud, але погано працює з продуктами Google.

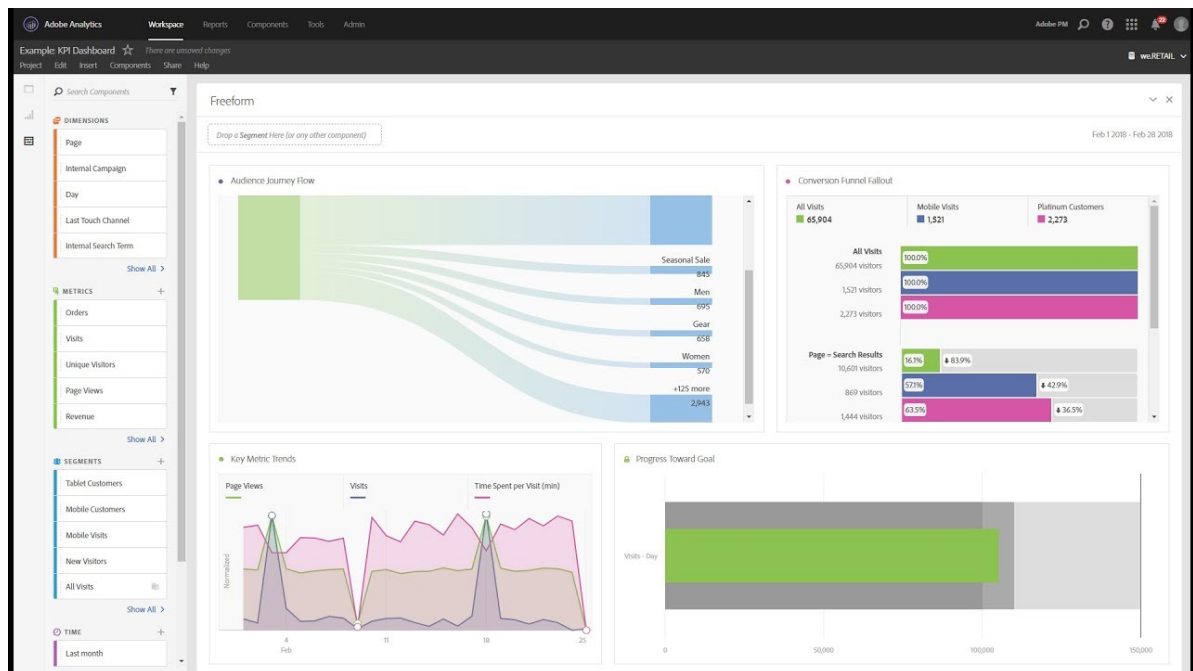


Рис. 3 – Сервіс Adobe Analytics

3.4 Сервіс Clicktale

ClickTale – це сервіс для аналізу веб-сайтів, який допомагає власникам веб-сайтів відстежувати та аналізувати поведінку користувачів. Він дозволяє користувачам отримувати інформацію про те, як відвідувачі навігуються їх веб-сайтами (рис. 4).

Переваги: ClickTale використовує теплокарти та повторні сеанси. Ці функції дозволяють отримати візуальне сприйняття поведінки відвідувачів. Користувач також може користуватися платформою в мобільній версії. Надаються різноманітні звіти, зокрема аналіз конверсій. Існують звіти, які пропонують практичну інформацію про те, як треба модифікувати функціональність веб-сайту.

Недоліки: у ClickTale доступно багато даних, але для початківців, це може не дати змогу зрозуміти бізнес-процеси. Необхідно самостійно аналізувати теплокарти, воронки переходів або записи відвідувачів. Також не вистачає можливості створювати звіти з використанням кількісних даних.

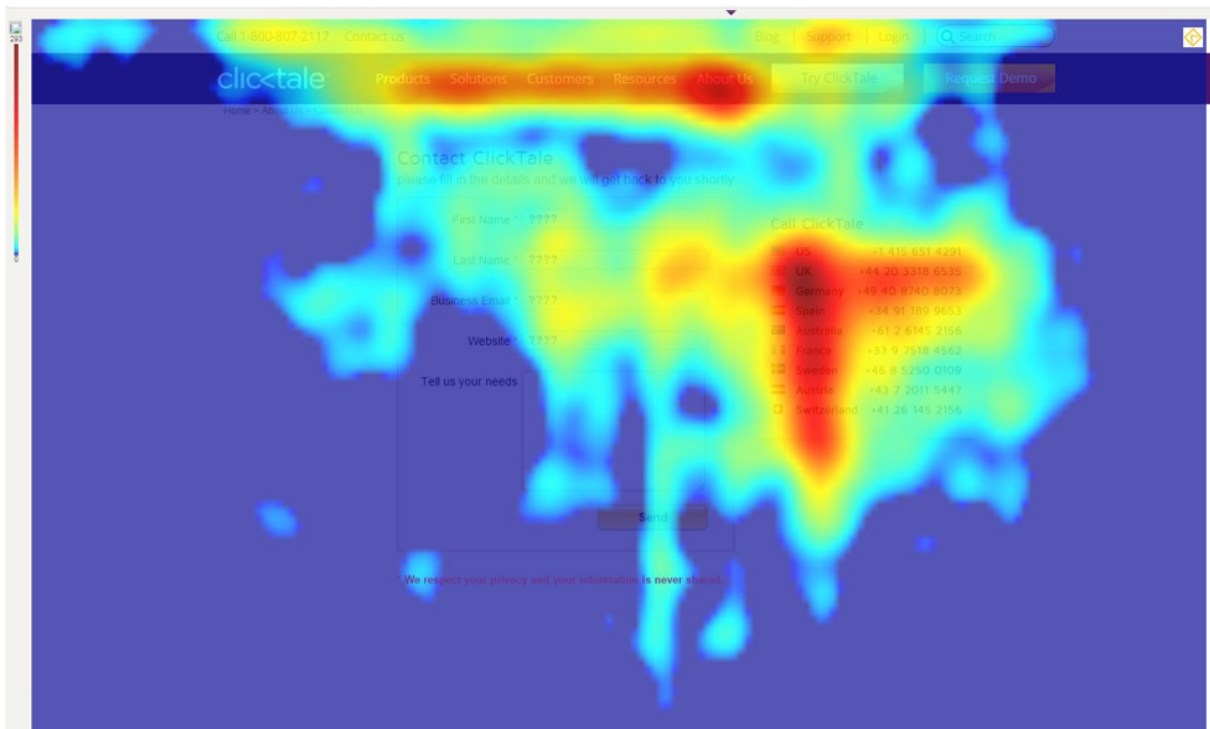


Рис. 4 – Теплокарти Clicktale

3.5 Порівняння

Існує перелік ключових факторів, які слід враховувати, вибираючи сервіс для веб-аналітики. У таблицях 1, 2 наведено порівняння вищезазначених сервісів для веб-аналітики.

Таблиця 1 – Підтримка платформ, соціальної аналітики та звітів сервісами веб-аналітики

| Сервіс | Підтримка платформ | Соціальна аналітика | Персоналізований звіт / Інфопанель |
|-------------------------|-------------------------------|--|--|
| Google Analytics | Всі ОС та браузерери | Звіти соціальної аналітики графі, реферали, Data Hub активність | Легко ділитися власними звітами. На інформаційних панелях використовуються віджети для швидкої та простої персоналізації, КПЕ тощо |
| Adobe Analytics | Всі ОС та браузерери окрім ІЕ | Adobe Social – управління соціальним вмістом та активністю в багатьох соціальних мережах та на сторінках профілю | Аналітичні звіти в режимі реального часу . Можливість створення інформаційних панелей та їх обміну |
| Click Tale | Всі ОС | Моніторинг соціальних мереж | Не персоналізований аналіз в реальному часі та візуалізація даних |

Таблиця 2 – Ліміти даних та можливість тестування для відомих сервісів веб-аналітики

| Сервіс | Ліміт даних | Тестування |
|-------------------------|---|----------------------|
| Google Analytics | Стандартна версія – 10 мільйонів викликів / місяць, 500 викликів / сеанс, 200,000 викликів / користувача / день. Преміум – 1 мільярд викликів / місяць | А/Б чи Багатовимірне |

Продовження таблиці 2

| Сервіс | Ліміт даних | Тестування |
|------------------------|--|---|
| Adobe Analytics | Не має щомісячного ліміту звернень | Шляхом нашарування додаткового продукту (Test & Target) |
| Click Tale | Безкоштовна версія – 5,000 переглядів сторінок/місяць. Золота версія – 10 доменів, 3,000,000 переглядів сторінок / місяць | А/Б чи Багатовимірне |

РОЗДІЛ 4: СИСТЕМА ВЕДЕННЯ СТАТИСТИКИ ВІДВІДУВАНЬ ВЕБ-САЙТІВ

4.1 Система «Website Visit Statistics»

В рамках дипломної роботи було розроблено систему відстеження відвідувачів сайту «Website Visit Statistics» (WVS). Система може збирати дані про відвідування та відображати наступні метрики:

- Кількість відвідувань, зокрема унікальних (гістограма):
 - за дні останнього тижня;
 - за дні останнього місяця;
 - за години останнього дня.
- Кількість відвідувань з певного пристрою (кругова діаграма):
 - за останній тиждень;
 - за останній місяць;
 - за визначений інтервал.
- Кількість відвідувань для країни (кругова діаграма та карта світу).
- Кількість відвідувань з певного браузера (кругова діаграма).
- Кількість відвідувань певної сторінки (кругова діаграма).
- Реферальні веб-сайти (кругова діаграма).
- Статистика для А/Б тестування (кругова діаграма).
- Середня, мінімальна, максимальна кількість відвідувань в день за визначений період часу.
- Загальна кількість активних користувачів за вказаний проміжок часу.
- Стандартне відхилення кількості користувачів за днями.

4.2 Огляд архітектури

4.2.1 Діаграма системи

На рис. 5 можна побачити користувачів і компоненти системи та їх взаємодію між собою. Адміністратор веб-сайту реєструє веб-сайт, використовуючи консоль WVS. Далі він налаштовує інтеграцію з аналітичною системою через програмний код свого веб-сайту. Цільовий веб-сайт передає дані про відвідування на API після завантаження веб-сторінки відвідувачем.

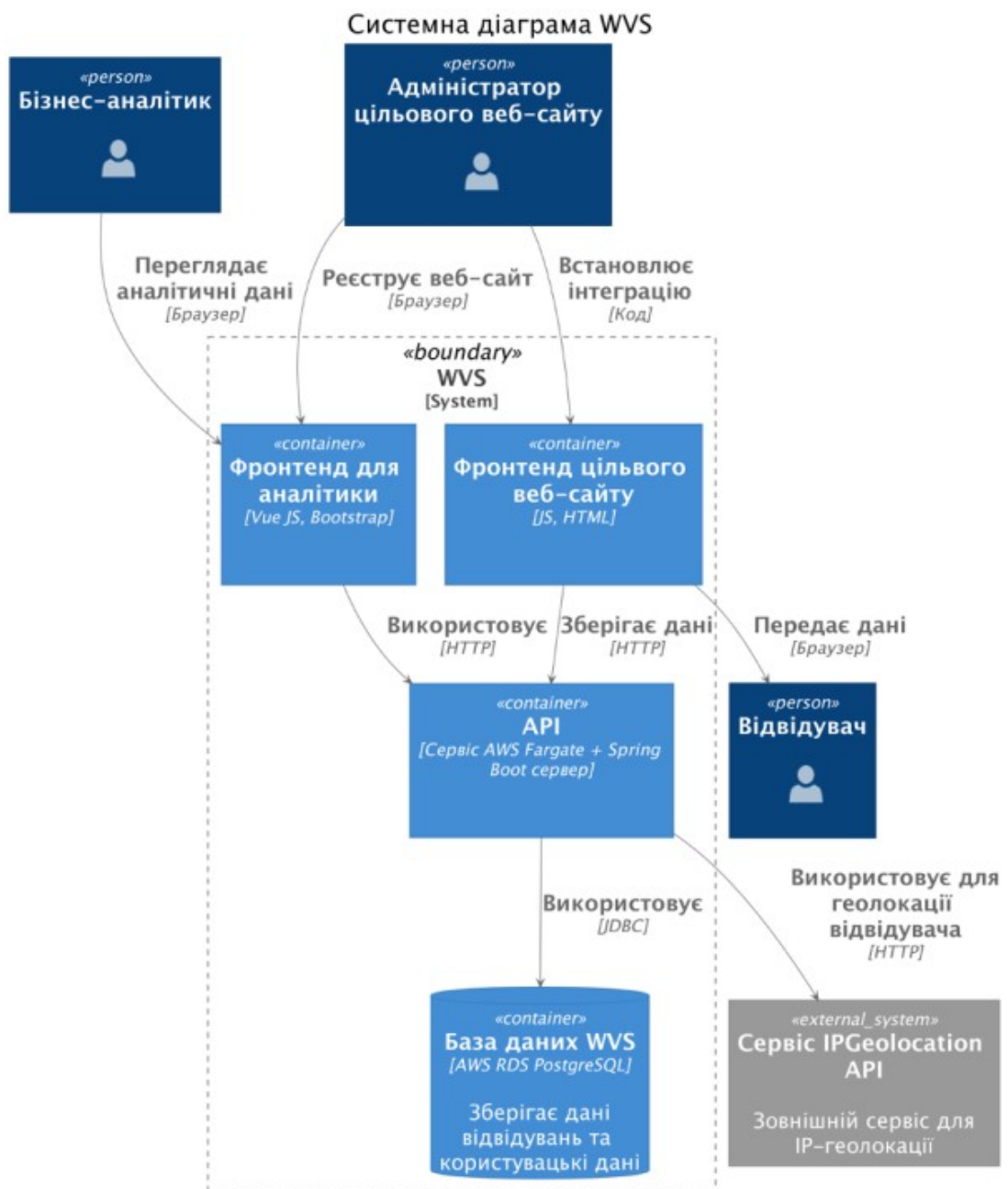


Рис. 5 – Архітектура WVS

Далі дані з серверу зберігаються в реляційну базу даних. Сервер запитує дані про геолокацію з зовнішнього сервісу перед збереженням. Після цього сервер аналізує дані про відвідування. Нарешті, бізнес-аналітик заходить на веб-сайт через браузер і переглядає звіти і діаграми про відвідування цільового веб-сайту. При цьому браузер запитує необхідні дані з REST API використовуючи HTTP протокол.

Існує два основних технічних способи збору даних. При використанні першого, традиційного методу аналізу файлів журналу сервера, відбувається зчитування журналів, в які веб-сервер записує запити на файли браузерами. У дипломній роботі було використано другий метод: позначення сторінок (page tagging).

Даний метод використовує JavaScript, вбудований у веб-сторінку, для надсилання запитів на виділений для аналітики сервер, коли веб-браузер відображає веб-сторінку. Скрипт отримує доступ до інформації браузера та передає її серверу через HTTP-запит. Сервер аналітики може бачити IP-адресу відправника в TCP-сегменті [12]. Це унікальна числова мітка, присвоєна кожному пристрою, підключеному до комп'ютерної мережі, яку використовує Інтернет-протокол.

4.2.2 Геолокація

Усі підключені до Інтернету пристрої мають IP-адресу, і більшість із них мають зворотній DNS запис. Зворотний DNS можна використати, щоб визначити, які компанії відвідують веб-сайт на основі їх IP-адрес. Якщо ввести IP-адресу в програму rDNS, вона повідомить про назву домену і допоможе провести геолокацію користувача. Це можна порівняти з пошуком номера телефону в адресній книзі, але для веб-сайтів (рис. 6).

При використанні rDNS, IP-адреса обертається, і до кінця додається **in-addr.arpa**. Наприклад, якщо маємо IPv4-адресу **87.125.196.129**, після застосування процедури rDNS, вона стане **129.196.125.87.in-addr.arpa**. Цей метод використовує PTR DNS-запис, що асоціює доменне ім'я з IP-адресою [13].

Прив'язка IP-адрес до геолокацій може здійснюватися за допомогою певних таблиць [14]. Однак цей метод не є точним. Основним джерелом даних про IP-адреси є Регіональний реєстр Інтернету (RIR), що розподіляє та розповсюджує IP-адреси між організаціями, розташованими у відповідних регіонах обслуговування. Для максимальної ефективності сервіси геолокації зазвичай використовують і rDNS, і дані цих таблиць.



Рис. 6 – Робота rDNS

4.3 Інтерфейс користувача

4.3.1 Автентифікація

Неавтентифікованого користувача система автоматично направляє на логін сторінку (рис. 7), де він може ввести ім'я та пароль користувача і увійти в систему. Якщо користувач є новим, він може зареєструватися (рис. 8), натиснувши кнопку «Sign Up», та заповнивши необхідні поля. Також передбачена можливість виходу з системи після натискання кнопки «Logout».

Enter username:

Enter password:

[Login](#) [Sign up](#)

Рис. 7 – Форма входу користувача (Login)

Enter username:

Enter email:

Enter full name:

Enter password:

[Sign up](#)

Рис. 8 – Форма реєстрації користувача (Sign Up)

4.3.2 Управління веб-сайтами

Щоб отримати статистики для веб-сайту спочатку необхідно його зареєструвати (рис. 9). Після того, як сайт зареєстровано, можна переглянути інформацію про нього (рис. 10).

New Site

Name:

Link:

[Create](#)

Рис. 9 – Реєстрація сайту (Site -> New Site)

Website Visit Statistics Console

Site ▾ Time Agent ▾ Referrals A/B Pages Summary Country ▾ Documentation
[Logout](#)

Current site

Name: **test-site**

Link: [link](#)

[Modify](#)

Copy these lines into header of your website:

```

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js" />
<script src="http://localhost:8787/js/visit.js" />
<script>
  let apiKey = '6a36d434-146b-4457-8d81-d7e6b71bdaf5';
  visit(apiKey, 'http://localhost:8787')
</script>

```

[Delete](#)

© All rights reserved.
Website Visit Statistics

Рис. 10 – Інформація про веб-сайт (Site -> Current Site)

Знизу можна побачити фрагмент коду. Його треба додати до тегу «head» цільового веб-сайту, і система зможе почати збирати статистику відвідувань. Є можливість редагувати інформацію про сайт. Після натискання кнопки «Modify» сторінка перейде в режим редагування (рис. 11).

Current site

Name:

Link:

Рис. 11 – Режим редагування сайту (Site -> Current Site)

Веб-сайт також можна видалити, якщо він не використовується. Існує багато статистичних даних, які WVS може візуалізувати. Вони поділені на декілька категорій: час відвідувань, агент користувача, країни, сторінки, А/Б тестування та реферальні веб-сайти.

4.3.3 Часові статистики

Однією з найважливіших метрик веб-сайту є кількість відвідувань. Вона може показати, як популярність сайту змінюється протягом часу. WVS надає користувачеві три часові показники (рис. 12 - 14).

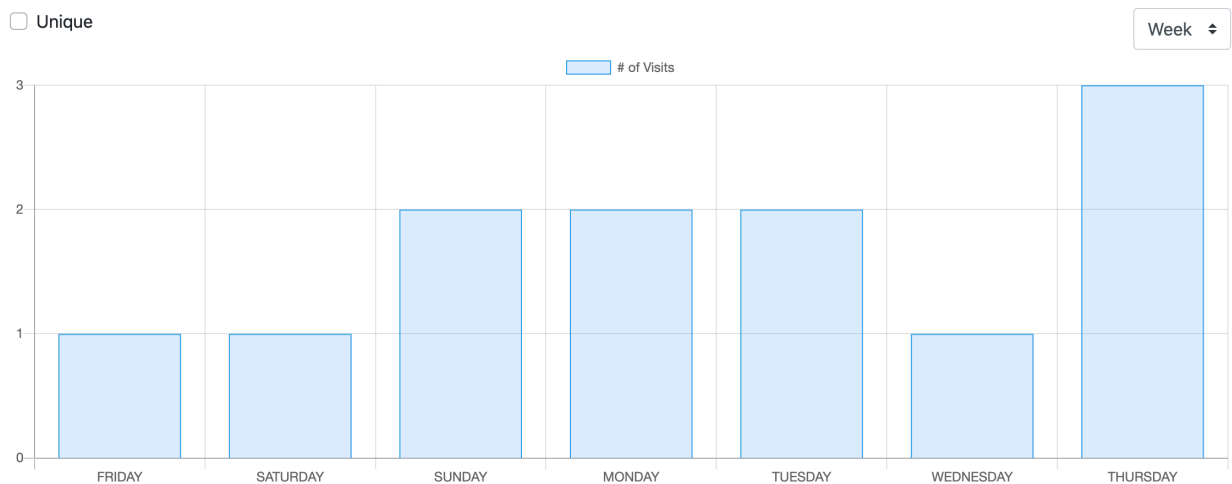


Рис. 12 – Кількість відвідувань в певний день за останній тиждень (Time)

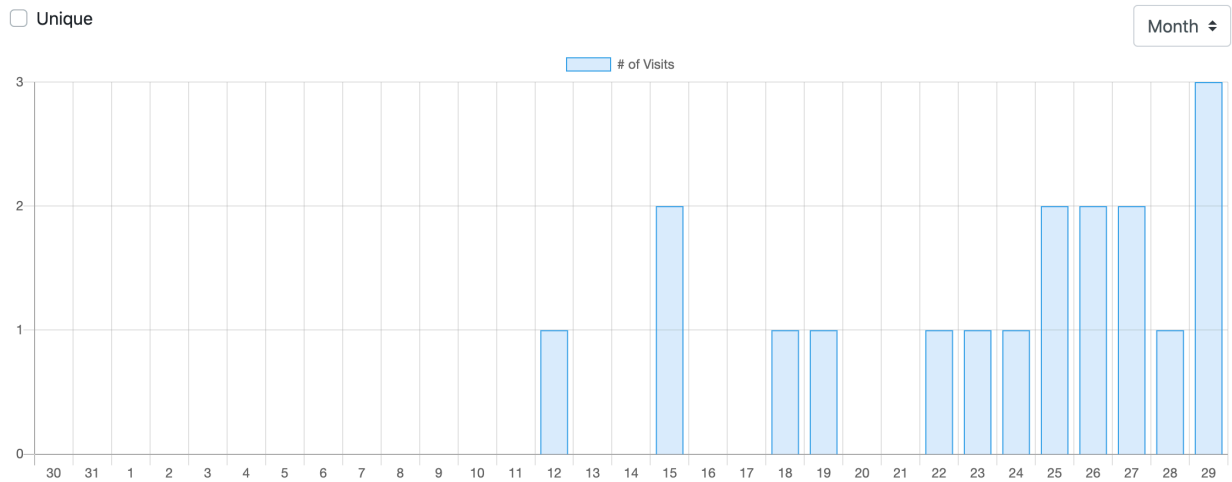


Рис. 13 – Кількість відвідувань в певний день за останній місяць (Time)

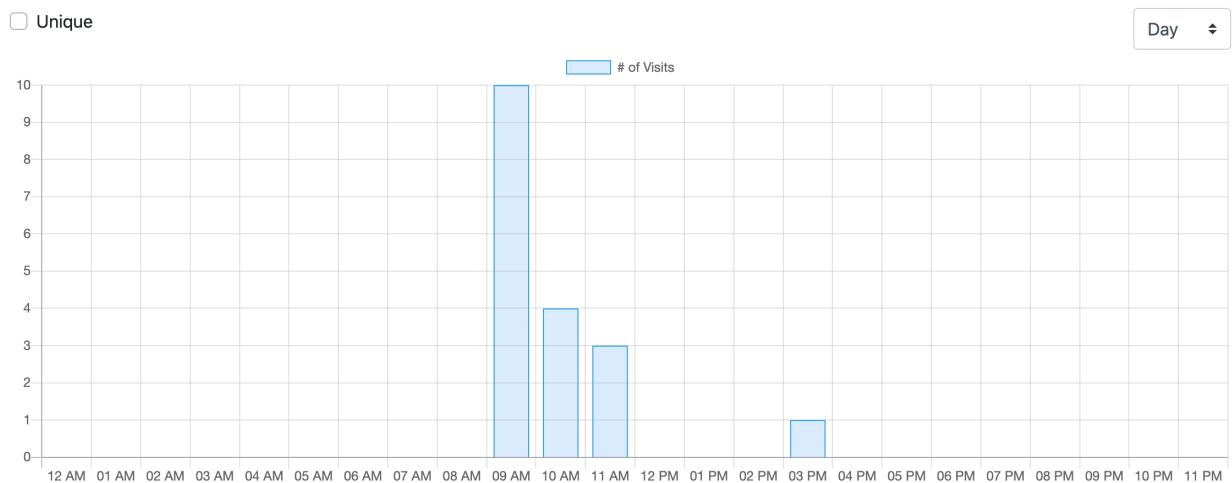


Рис. 14 – Кількість відвідувань в певну годину за останній день (Time)

Остання діаграма (рис. 14) допомагає зрозуміти, в якій частині дня сайт є більш відвідуваним. Наприклад, якщо існує стримінговий сервіс, вона вказуватиме години пік, коли можна зібрати більше аудиторії.

Є можливість показувати тільки унікальні відвідування, якщо ввімкнути прапорець «Unique». Таким чином, буде зрозуміло реальний розмір аудиторії веб-сайту. Графіки оновлюються автоматично при зміні часового діапазону чи при виборі унікальності відвідувань.

4.3.4 Статистики програмного агента

Сьогодні майже всі веб-сайти мають версії як для ПК, так і для мобільних телефонів. Припустимо, що бізнес хоче дізнатися, яка частка користувачів

використовують мобільну версію. Щоб визначити це, стане у нагоді метрика кількості відвідувань для певного пристрою (рис. 15). Доступні 3 часові інтервали: для останнього тижня, місяця та вибраний інтервал.

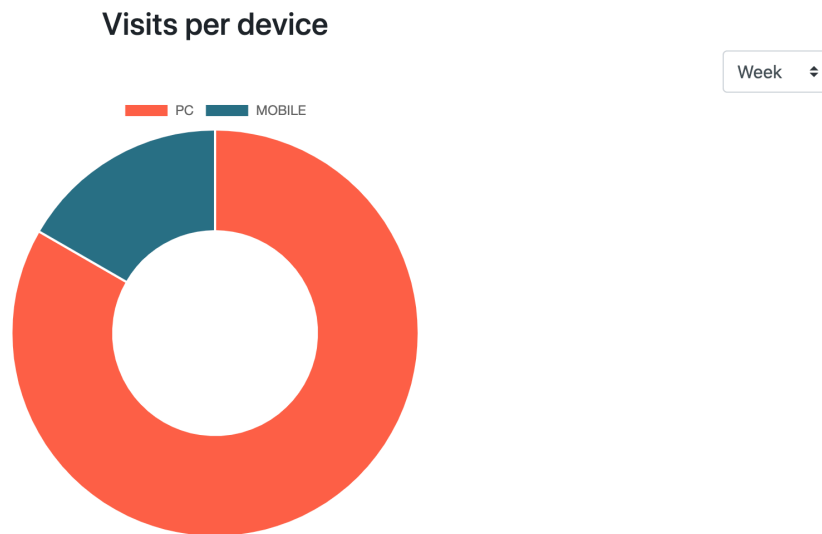


Рис.15 – Кількість відвідувань за пристроєм (Agent ->Device)

Досить корисно знати, якими веб-браузерами користуються відвідувачі веб-сайту. Адже відображення веб-сторінки може залежати від типу браузера і деяка функціональність може не підтримуватись. Тому потрібно докладати деяких зусиль, щоб зробити веб-сайти мульти-браузерними.

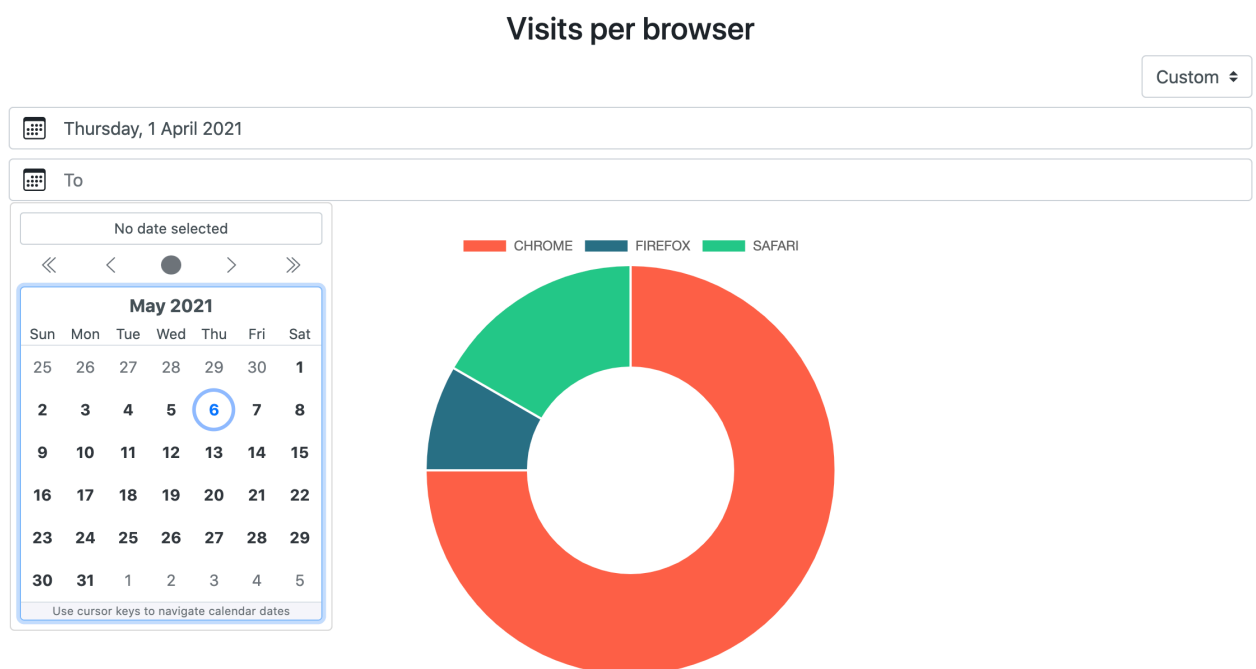


Рис. 16 – Кількість відвідувань за веб-браузером (Agent ->Device)

Метрика кількості відвідувань за веб-браузером показує, на які браузери необхідно адаптувати візуальний інтерфейс цільового веб-сайту (рис. 16). На даному рисунку, зокрема, можна побачити діалогове вікно для вибору власного інтервалу дат, що надає певної гнучкості.

4.3.5 Статистики країн

Іноді важливо знати, користувачі з яких країн переглядають сторінки веб-сайту. Нехай, є інтернаціональний сайт і його можна перекласти на деякі нові мови. Треба визначити ці мови, або зробити висновок, що це не варте зусиль, необхідних для цього. Метрика кількості відвідувань за країною допоможе в цій ситуації. Як і попередні показники, вона підтримує 3 часові інтервали: останній тиждень, місяць та власний інтервал. Додатково до цього доступні 2 типи діаграм (рис. 17 - 18).

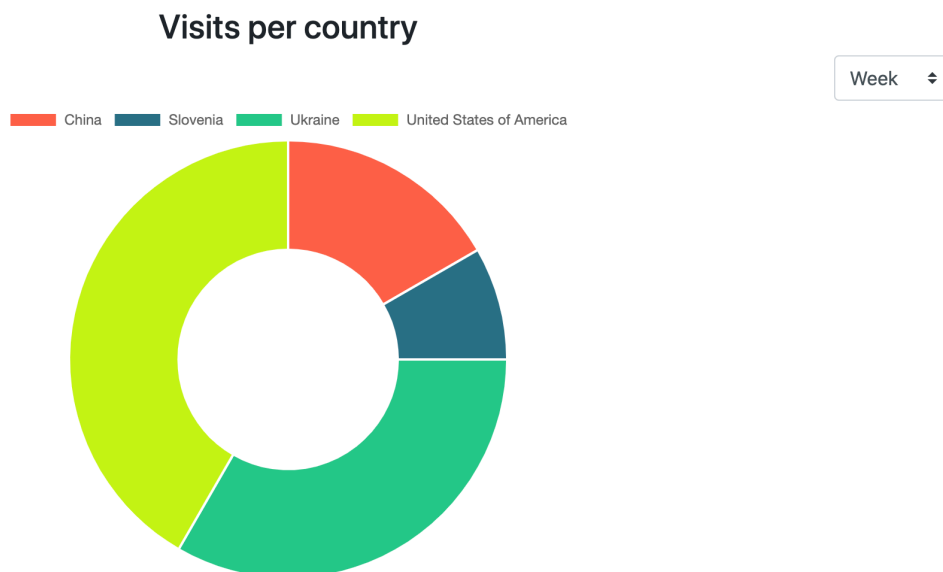


Рис. 17 – Кількість відвідувань за країною, кругова діаграма (Country -> Chart)

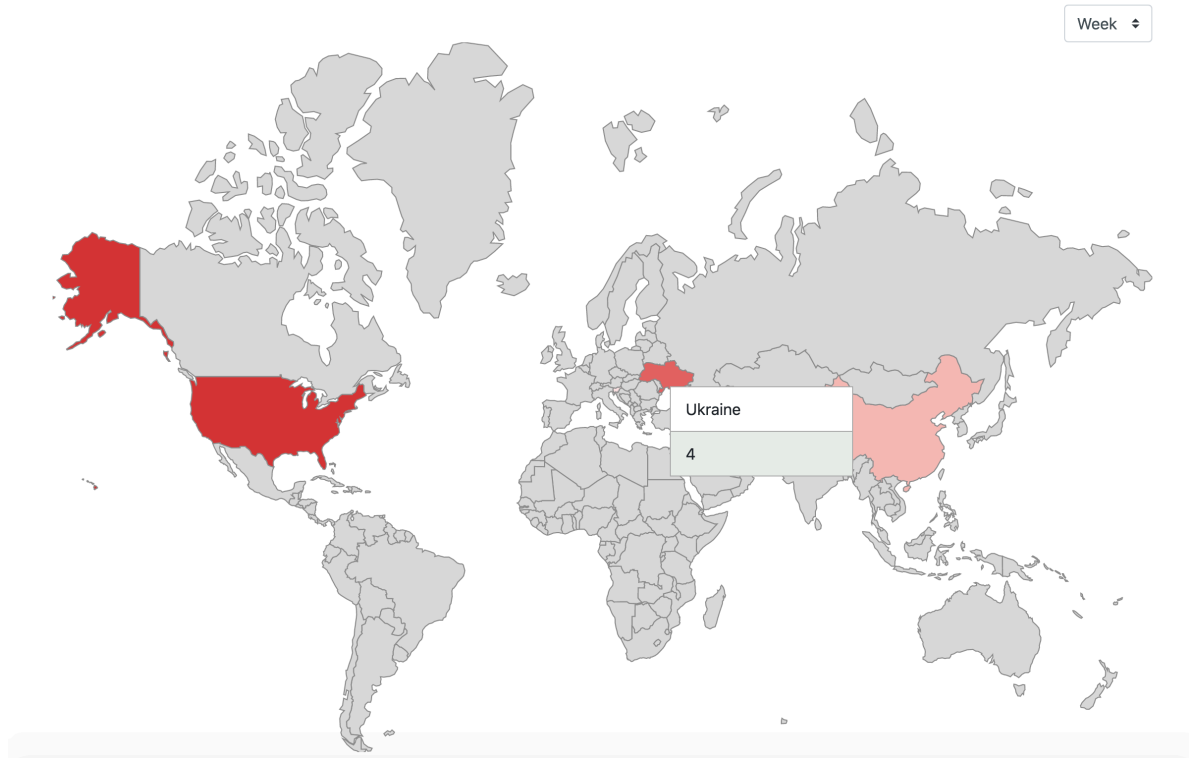


Рис. 18 – Кількість відвідувань за країною на карті світу (Country -> Map)

4.3.6 Статистики сторінок

Наступна статистика дає змогу знайти найбільш та найменш популярні сторінки деякого веб-сайту. Зазвичай, більш відвідуваним сторінкам треба приділяти більше часу.

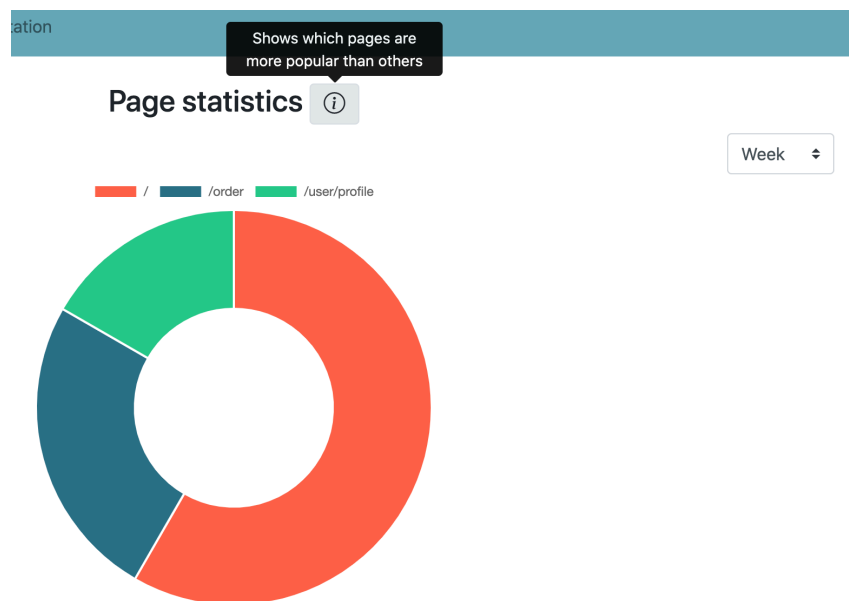


Рис. 19 – Кількість відвідувань за веб-сторінкою (Pages)

В той же час, проблемою менш відвідуваних веб-сторінок може бути нецікаве наповнення або складність в навігації до них. Наприклад, на рис. 19 можна побачити, що сторінка замовлень є більш популярною, ніж сторінка профілю користувача.

4.3.7 Реферальні веб-сайти

Реферальний веб-сайт – це ім'я сайту, що використовується для переходу відвідувача на цільовий веб-сайт, зазвичай через гіперпосилання. Тобто, реферальний веб-сайт є джерелом для трафіку. Це допомагає визначити, наскільки ефективною є реклама на деяких платформах. З рис. 20 видно, що найбільшим джерелом реферального трафіку даного веб-сайту є соціальна мережа Твітер. Проте більшість відвідувань таки відбуваються напряму.

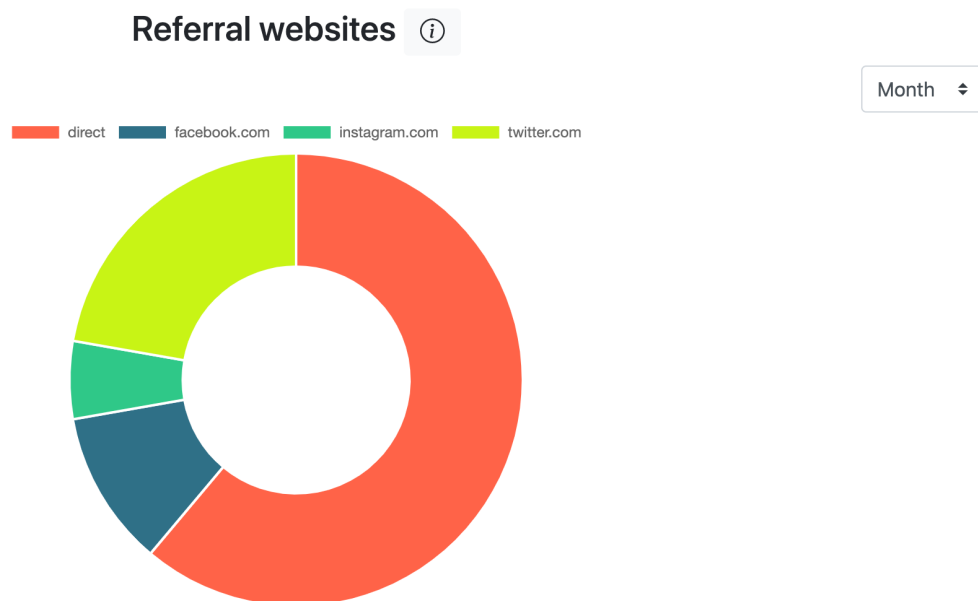


Рис. 20 – Кількість відвідувань з реферального веб-сайту (Referrals)

4.3.8 А/Б тестування

Одне з пристосувань веб-аналітики – А/Б тестування. Припустимо, користувач випускає нову версію інтерфейсу веб-сайту v2 27-го квітня. Передбачена можливість повернутися до старої версії. Тепер треба зрозуміти, чи достатньо хорошою є нова версія та чи є сенс залишати стару версію веб-сайту. Нехай сьогодні 3-є травня, тоді з кругової діаграми на рис. 21 можна зробити

висновок, що нова версія є досить успішною, проте варто залишити попередню, оскільки велика частка користувачів використовують цю версію.

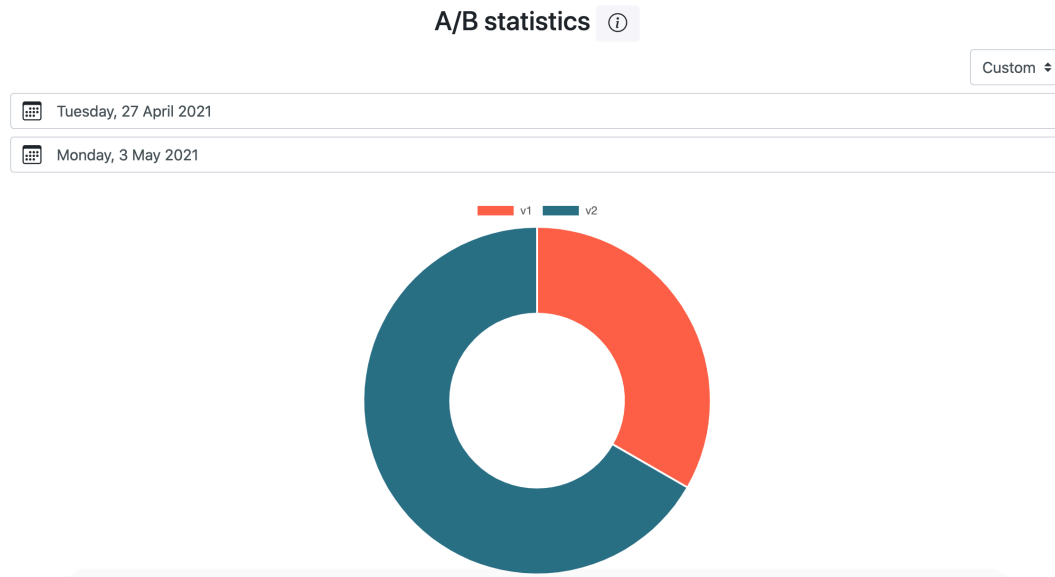


Рис. 21 – Кількість відвідувань за версією веб-сайту (A/B)

4.3.9 Загальна інформація

Система надає можливість переглядати загальні статистики у вигляді таблиці (рис. 22). Тут вказано кількість активних користувачів за певний період часу: середню, мінімальну та максимальну кількість відвідувань за день. Також розраховано стандартне відхилення відвідувань. Останній показник допомагає зрозуміти наскільки сильно відрізняється трафік в різні дні. Це може спонукати до побудови еластичної системи, яка збільшує свої потужності при збільшенні трафіку і навпаки. З іншого боку, якщо трафік досить постійний, веб-сайт може використовувати статичну кількість ресурсів. Додатково є можливість завантажити даний звіт у форматі Excel.

В консолі системи наявна документація для інтеграції нового веб-сайту. Також наведено кроки для використання функцій реферальних веб-сайтів та A/B тестування.

Visits summary per day

Export

Week ▾

| Name | Value |
|----------------------------------|-------|
| Total Active Users | 4 |
| Average Visits | 1.7 |
| Average Unique Visits | 1.4 |
| Minimum Visits | 1 |
| Minimum Unique Visits | 1 |
| Maximum Visits | 3 |
| Maximum Unique Visits | 3 |
| Visits Standard Deviation | 0.7 |
| Unique Visits Standard Deviation | 0.8 |

© All rights reserved.
Website Visit Statistics

visit_summary.xlsx

Show all ×

Рис. 22 – Загальна інформація про веб-сайт та експортування в Excel (Summary)

4.4 Деталі реалізації

4.4.1 Бекенд

Фреймворк (framework) для аналітичного серверу, використаний в цій роботі – **Spring Boot**. Він спрощує створення самостійних програм, основаних на Spring, які досить легко запустити. Більшість програм Spring Boot потребують мінімальної конфігурації Spring [15].

Spring Framework – це програмний фреймворк та контейнер з інверсію управління (IoC) для мови **Java**. Основні функції фреймворку можуть бути використованими будь-якими додатками Java, проте для створення веб-додатків для платформи Java EE (Enterprise Edition) необхідні відповідні розширення. Хоча фреймворк не нав'язує жодної конкретної моделі програмування, він став популярним у спільноті Java як доповнення або навіть заміна моделі Enterprise JavaBeans (EJB).

В якості бібліотеки для версіонування бази даних було обрано **Flyway**. Flyway – це бібліотека для міграції баз даних з відкритим кодом. Міграції можуть бути записані на мові SQL або Java. Вона має клієнт командного рядка, Java API

для міграції бази даних під час запуску програми, плагіни для **Gradle** та **Spring Boot**.

Для зберігання даних використано **PostgreSQL**, реляційну систему управління базами даних (СУБД), що підкреслює розширюваність та відповідність стандартам SQL. Побачити схему бази даних разом з таблицею Flyway History, що необхідна для міграції, можна на рис. 23:

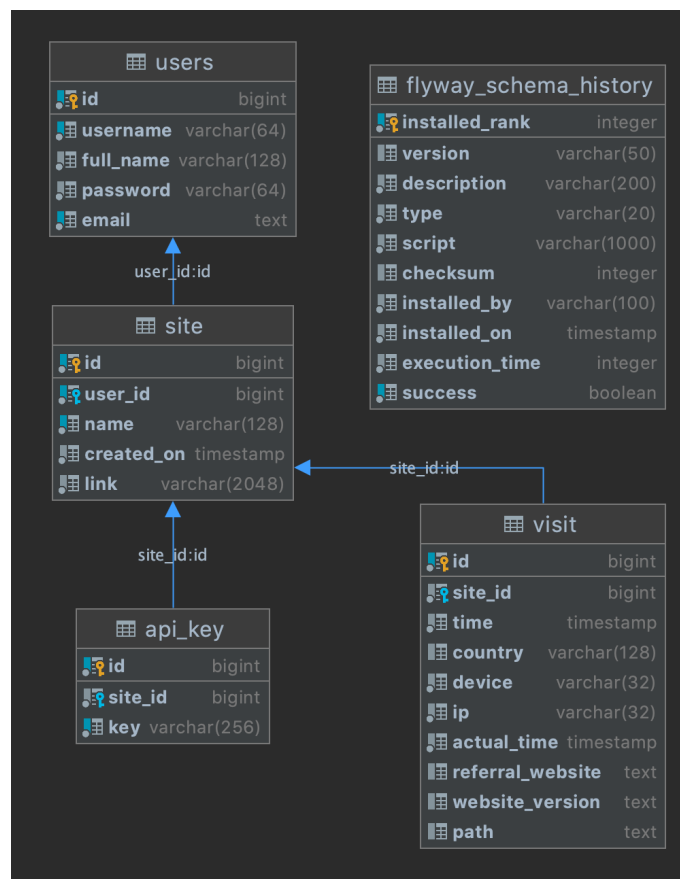


Рис. 23 – Схема бази даних

Користувач може мати багато сайтів. В сайту може бути багато відвідувань. Кожен сайт має унікальний ключ API. Щоб мати змогу розрахувати статистику відвідувань, зберігається наступний список полів в таблиці відвідувань:

- time – місцевий час відвідування;
- country – країна відвідувача (ISO 3166 2-символьний код країни);
- device – пристрій відвідувача (PC / MOBILE);
- ip – IP-адреса відвідувача;
- actual_time – час відвідування в UTC (Coordinated Universal Time);

- `referral_website` – реферальний веб-сайт;
- `website_version` – версія веб-сайту;
- `path` – сторінка веб-сайту.

Тип пристрою та браузер визначається за допомогою користувачького агента, переданого JavaScript за допомогою регулярного виразу [16]. Тим часом для пошуку країн використано службу **IpGeolocation**. IP Geo API надає інформацію про місцезнаходження будь-якої IP-адреси [17]. Для всіх часових статистик WVS використовує місцевий час. Хоча він не відображає точного моменту в часовій шкалі UTC, він представляє точку часової шкали зі сторони потенційного відвідувача [18].

4.4.2 Інтеграція

Щоб система могла збирати дані про відвідування веб-сайту, необхідно включити нижченаведені рядки до тегу «`head`» веб-сторінок. Повний приклад інтеграції наведений в додатку А. Після відкриття сторінки у браузері скрипт **visit.js** (додаток Б) надсилає інформацію про браузер разом із місцевим часом за допомогою HTTP запиту на аналітичний сервер. Для AJAX запиту була використана бібліотека **jQuery** (рис. 18) [19].

```
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
<script src="http://https://wvs-backend.andlvovsky.com/js/visit.js"></script>
<script>
  let apiKey = '6a36d434-146b-4457-8d81-d7e6b71bdaf5'; // write the api key
  for your site
  visit(apiKey, 'https://wvs-backend.andlvovsky.com', 'v2')
</script>
```

4.4.3 Фронтенд

Дані про відвідування для бізнес-аналітика не мають великої користі, поки вони не стануть візуалізованими. Тому, була створена консоль WVS, щоб мати змогу бачити статистику відвідувань у вигляді діаграм.

Як фронтенд-фреймворк обрано **Nuxt.js**. Nuxt.js – це безкоштовний фреймворк для веб-додатків з відкритим кодом, оснований на **Vue.js**, **Node.js**, **Webpack** та **Babel.js**. Він є мета-фреймворком для універсальних додатків [20].

Vue.js – це model-view-viewmodel **JavaScript** фреймворк з відкритим кодом для побудови користувацьких інтерфейсів та односторінкових (SPA) програм. Vue.js має архітектуру, що покроково адаптується до вимог і фокусується на декларативному стилі та композиції компонентів. Основна бібліотека орієнтована лише на шар view. Розширені функції, необхідні для складних додатків, такі як: маршрутизація, управління станом та інструменти побудови, пропонуються через офіційно підтримувані бібліотеки та пакети.

Для створення діаграм використовується бібліотека **Chart.js**, досить проста, але гнучка для дизайнерів та розробників. Зокрема, проект використовує бібліотеку **vue-chartjs**. Це дозволяє використовувати Chart.js без додаткових пристосувань всередині Vue. Вона добре підходить для тих, кому потрібна проста та швидка веб-графіка. Бібліотека абстрагується від базової логіки, але надає об'єкт Chart.js, щоб мати максимальну гнучкість [21].

Фреймворк CSS для WVS – це **Bootstrap**. Bootstrap – це CSS фреймворк, спрямований на адаптивну мобільну веб-розробку. Він містить шаблони дизайну на основі CSS та JavaScript для типографіки, форм, кнопок, навігації та інших компонентів інтерфейсу.

Основним плагіном (plug-in) CSS є **BootstrapVue**. Він дозволяє створювати адаптивні, мобільні та доступні веб-проекти за допомогою Vue.js та найпопулярнішої інтерфейсної бібліотеки CSS – Bootstrap v4 [22].

4.4.4 Інфраструктура

Для розгортання інфраструктури було обрано найбільш популярну платформу для хмарних обчислень **Amazon Web Services (AWS)** [23]. Визначальними факторами вибору були наявність гнучкості в конфігурації системи, безкоштовної підписки на один рік та надання багатьох різноманітних сервісів [24].

На рис. 24 зображена діаграма інфраструктури побудована за допомогою **PlantUML**, безкоштовного програмного засобу для створення діаграм за допомогою програмного коду [25]. В ролі сервісу домених імен використано

Route 53. Даний сервіс є високодоступним та масштабовним. На ньому створено 5 DNS записів (рис. 25). Перший (NS – запис) визначає перелік серверів імен, а другий (SOA – запис) містить адміністративну інформацію про зону хостингу. Дані записи були створені автоматично. Наступний (CNAME – запис) було створено для перевірки SSL сертифікату. Останні два A – записи містять доменні ім'я API та фронтенду відповідно. DNS запис для API вказує на IP-адресу балансувальника трафіку, а запис для фронтенду – на дистрибутив мережі розповсюдження контенту (CDN).

CloudFront CDN отримує статичні файли веб-сайту з сервіса-сховища даних **S3**. Даний CDN сервіс дає можливість отримати необхідні для завантаження веб-сайту дані в будь-якій точці світу за дуже короткий час.

Розроблена веб-система потенційно може стикнутися з великим обсягом веб-трафіку, тому вкрай важливо, щоб ця система була масштабовною. Також необхідно забезпечити відмовостійкість та доступність системи. Для цього в нагоді став сервіс **Fargate**. Це безсерверна обчислювальна платформа для контейнерів. Вона надає можливість автоматично збільшувати кількість серверів при збільшенні трафіку і навпаки. Також, якщо сервіс побачить, що якийсь з серверів не працює, він буде зупинений і на його місці запуститься новий сервер.

При використанні безсерверних рішень розробникам не треба мати справу з налаштуванням серверів для запускання коду. Проте треба розуміти, що сервіс який запускаються не повинен зберігати стан (повинен бути stateless), адже сервери можуть бути зупинені в будь-який момент часу.

У якості служби моніторингу **Fargate** використовує **CloudWatch**. Цей сервіс збирає з серверів журнальні файли та метрики, такі як відсоток використаного процесора, використаної оперативної пам'яті тощо. Він допомагає бачити ресурсну ефективність та налагоджувати систему.

Всі дані про відвідування веб-сайтів та користувачів зберігаються в реляцій базі PostgreSQL, що управляється сервісом **RDS**. Даний сервіс автоматично проводить оновлення програмного забезпечення та надає можливість для автоматичного та ручного резервного копіювання.

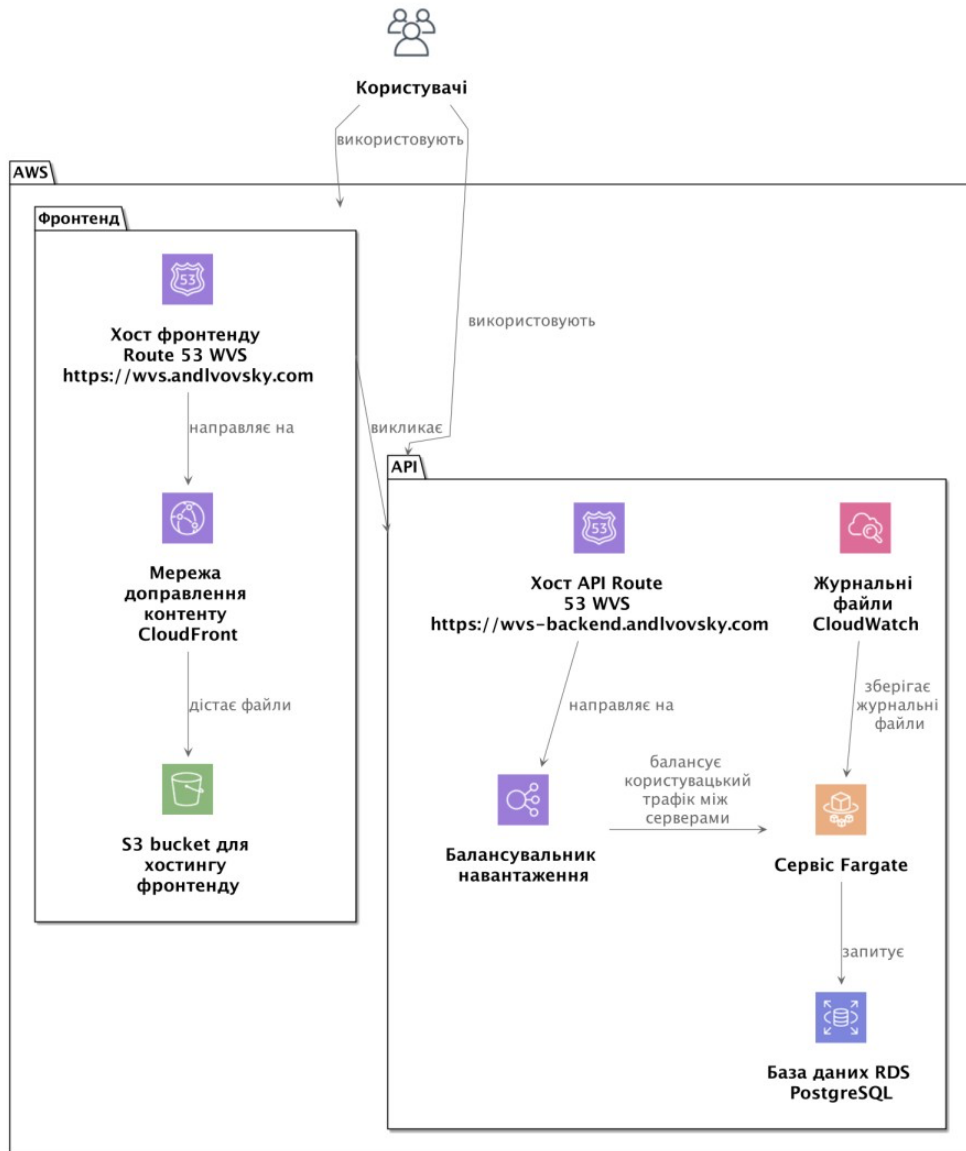


Рис. 24 – Діаграма AWS інфраструктури

| Record name | Type | Routin... | Differ... | Value/Route traffic to |
|--------------------------------|-------|-----------|-----------|--|
| andlvovsky.com | NS | Simple | - | ns-389.awsdns-48.com. ns-980.awsdns-58.net. ns-1968.awsdns-54.co.uk. ns-1152.awsdns-16.org. |
| andlvovsky.com | SOA | Simple | - | ns-389.awsdns-48.com. awsdns-hostmaster.amazon.com. 1 7200 900 1209600 86400 |
| _a2c9e9efa066c2e91da8463265... | CNAME | Simple | - | _eb3ccc947806eef88732ba6fbf7bb9d8.zzzl1nslwt.acm-validations.aws. |
| wvs-backend.andlvovsky.com | A | Simple | - | dualstack.wvs-lb-1523165624.us-east-1.elb.amazonaws.com. |
| wvs.andlvovsky.com | A | Simple | - | d2043n05npqgy.cloudfront.net. |

Рис. 25 – DNS записи на сервісі Route 53

Для розгортання частини інфраструктури використано **Terraform** – засіб для опису інфраструктури в програмному коді [26]. Користувачі описують

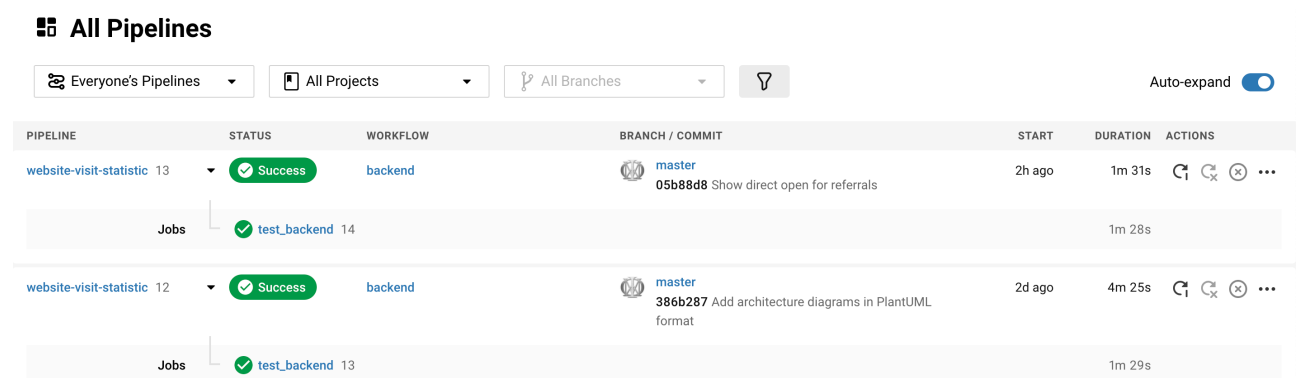
інфраструктуру в декларативному стилі, використовуючи мову HCL (HashiCorp Configuration Language). Далі наведено модуль для створення S3 bucket-у (аналог кореневої папки) необхідний для зберігання статичних файлів веб-сайту. Цей, досить стислий файл, знімає необхідність створювати даний ресурс вручну, використовуючи консоль AWS. Загалом, опис інфраструктури в кодї (IaC) є досить корисним підходом, оскільки таку систему можна розгортати та згорнути в будь-який момент і не треба запам'ятовувати всю необхідну конфігурацію.

```
resource "aws_s3_bucket" "website" {
  bucket = var.website_bucket_name
  acl    = "public-read"

  website {
    index_document = "index.html"
    error_document = "index.html"
  }
}
```

4.4.5 Неперервна інтеграція

У якості сервісу для неперервної інтеграції обрано, одне за найбільш популярніших рішень – **CircleCI** [27]. Було налаштовано виконання автоматизованих юніт (unit) та інтеграційних (integration) тестів при кожному відправленні нового програмного коду в систему керування версіями **Github**. На рис. 26 можна побачити успішне виконання тестів для розробленої системи. Використаний конфігураційний файл наведено в додатку В.



The screenshot shows the CircleCI 'All Pipelines' view. It displays a table of pipeline runs with columns for Pipeline, Status, Workflow, Branch / Commit, Start, Duration, and Actions. Two pipeline runs are visible, both with a 'Success' status. The first run is for commit '05b88d8' and the second for '386b287'. Each pipeline run includes a 'Jobs' section with a 'test_backend' job that completed successfully.

| PIPELINE | STATUS | WORKFLOW | BRANCH / COMMIT | START | DURATION | ACTIONS |
|----------------------------|---------|-----------------|--|--------|----------|---------|
| website-visit-statistic 13 | Success | backend | master 05b88d8 Show direct open for referrals | 2h ago | 1m 31s | 🔄 🗑️ ⋮ |
| Jobs | | test_backend 14 | | | 1m 28s | |
| website-visit-statistic 12 | Success | backend | master 386b287 Add architecture diagrams in PlantUML format | 2d ago | 4m 25s | 🔄 🗑️ ⋮ |
| Jobs | | test_backend 13 | | | 1m 29s | |

Рис. 26 – Результати виконання тестів на CircleCI

4.4.6 Розгортання нової версії системи

Нижче наведено скрипт на мові bash **deploy-backend.sh** для розгортання нової версії API. Спочатку відбувається вхід в сховище для докер-образів. Далі

програма збирається і поміщається в докер-образ. Після цього даний образ завантажується у відповідне сховище. Останнім кроком визивається оновлення сервісу **Fargate**.

```
# login to ECR
aws ecr-public get-login-password --region us-east-1 | docker login --username AWS
--password-stdin public.ecr.aws/u2x1f6d4
# build application
./gradlew clean build -x test
# build image
docker build -t wvs .
# tag image
docker tag wvs:latest public.ecr.aws/u2x1f6d4/wvs:latest
# push image
docker push public.ecr.aws/u2x1f6d4/wvs:latest
# update ECS service
aws ecs update-service --cluster wvs --service main --force-new-deployment
```

Процес розгортання фронтенду є дещо простішим (**deploy-frontend.sh**). Спочатку відбувається компіляція програмного коду в статичні файли, що зберігаються в папку **dist**. Наступним кроком дані файли завантажуються на хмарне сховище **S3**.

```
# build static files
NEXT_ENV_BACKEND_URL=https://wvs-backend.andlvovsky.com npm run build
# upload static files to S3
aws s3 sync dist s3://website-visit-statistics --delete --acl public-read --cache-control no-cache
```

4.4.7 Локальний запуск системи

Для того, щоб побудувати бекенд, треба перейти до директорії **backend**. Потім необхідно використати наступні команди для побудови та запуску:

```
./gradlew clean build -x test
./gradlew bootRun
```

Для роботи з фронтендом треба перейти до каталогу **frontend**. Після цього необхідно виконати наступні команди для побудови та запуску:

```
npm ci
npm run dev
```

Консоль WVS буде працювати на порту 3535. Щоб запустити базу даних, можна або встановити Postgres 9.6.1 і створити базу даних вручну, або запустити її в контейнері Docker. База даних повинна мати ім'я **wvs** і працювати на порту 5435.

```
docker run -p 5434:5432 -e POSTGRES_USER=postgres \  
-e POSTGRES_PASSWORD=password \  
-e POSTGRES_DB=wvs -d postgres:9.6.1
```

Для запуску проекту на комп'ютері повинен бути встановлений певний набір програм. Він складається з:

- JDK 1.11 (Java Development Kit);
- NPM 6.14.5 (Node Package Manager);
- Docker 20.10.5.

ВИСНОВКИ

У роботі було досліджено поняття аналітики веб-сайтів та відстеження активності користувачів веб-сайтів. Продемонстровано типові способи збору даних про відвідування та визначено основні методи для цього. Проаналізовано популярні сервіси для відстеження дій користувачів на сайті. Зроблено огляд різних веб-відстежувачів, зокрема вбудованих скриптів. Показано взаємозв'язок поведінкової аналітики з веб-аналітикою. У результаті порівняння сервісів для веб-аналітики зроблено висновок, що найпопулярнішим інструментом веб-аналітики наразі є Google Analytics. Але існують потужні конкуренти, такі як Adobe Analytics та Clicktale, які забезпечують унікальну взаємодію з користувачами. Перший є дуже персоналізованим, а другий надає користувачам сторінку з теплокартами.

В рамках роботи розроблена система ведення статистики відвідувань веб-сайтів. Показано способи побудови системи для веб-аналітики з використанням фреймворків з відкритим кодом, бібліотек та сервісів. Систему було розгорнуто в платформі для хмарних обчислень із забезпеченням масштабовності та відмовостійкості. Налаштовано процес неперервної інтеграції. Система використовує технологію позначення сторінок для збору даних відвідувача. Було продемонстровано, як розроблений сервіс може інтегруватися з існуючими веб-сайтами та починати збирати дані про відвідування. Для візуалізації метрик було розроблено користувальницьку консоль, яка може показувати багато корисних діаграм та звітів, що можна розділити на групи.

В майбутньому планується розробка мобільної версії клієнта для реалізованої системи. А також тестування на відмовостійкість та тестування навантаженням, щоб запобігти потенційним проблемам за наявності великого об'єму веб-трафіку.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. A brief history of web analytics [Електронний ресурс]:
<https://www.clicktale.com/resources/blog/a-brief-history-of-web-analytics/>
2. Siroker D. A / B Testing: The Most Powerful Way to Turn Clicks Into Customers / D. Siroker. – К.: «Wiley», 2015. – 208 с.
3. Web Analytics [Електронний ресурс]: <https://www.optimizely.com/optimization-glossary/web-analytics/>
4. Web Analytics Overview [Електронний ресурс]:
https://www.researchgate.net/publication/272815693_Web_Analytics_Overview
5. Leveraging Advertising Data For Behavioral Insights [Електронний ресурс]:
<https://martech.org/leveraging-advertising-data-for-behavioral-insights/>
6. Web tracking [Електронний ресурс]: <https://koofr.eu/blog/posts/what-is-online-tracking-and-how-do-websites-track-you>
7. No-judgment digital definitions: What is a web tracker? [Електронний ресурс]:
<https://blog.mozilla.org/firefox/what-is-a-web-tracker/>
8. Web Tracking: What You Should Know About Your Privacy Online [Електронний ресурс]: <https://www.freecodecamp.org/news/what-you-should-know-about-web-tracking-and-how-it-affects-your-online-privacy-42935355525/>
9. How do websites track users? Technologies and methods, CCPA and GDPR compliance [Електронний ресурс]: <https://www.cookiebot.com/en/website-tracking/>
10. 11 Best Web Analytics Tools [Електронний ресурс]:
<https://www.inc.com/guides/12/2010/11-best-web-analytics-tools.html>
11. Web analytics vendors review & comparison sheet: which solution will be right for you? [Електронний ресурс]: <https://piwik.pro/blog/web-analytics-vendors-comparison-sheet/>
12. Spring MVC – How to get client IP address [Електронний ресурс]:
<https://mkyong.com/spring-mvc/spring-mvc-how-to-get-client-ip-address/>
13. rDNS [Електронний ресурс]: <https://www.liquidweb.com/kb/reverse-dns-lookup/>
14. Anthony T. HTML5 Geolocation 1st edition / T. Anthony. – К.: «O'Reilly Media», 2011. – 103 с.

15. Spring Boot Reference Documentation [Электронный ресурс]:
<https://docs.spring.io/spring-boot/docs/2.3.0.RELEASE/reference/html/>
16. What is the best way to detect a mobile device? [Электронный ресурс]:
<https://stackoverflow.com/questions/3514784/what-is-the-best-way-to-detect-a-mobile-device>
17. ipgeolocation API portfolio [Электронный ресурс]:
<https://ipgeolocation.io/documentation.html>
18. LocalDateTime to ZonedDateTime [Электронный ресурс]:
<https://stackoverflow.com/questions/36417317/localdatetime-to-zoneddatetime>
19. jQuery API [Электронный ресурс]: <https://api.jquery.com/>
20. Nuxt.js. Introduction [Электронный ресурс]: <https://nuxtjs.org/guide>
21. vue-chartjs. Getting started [Электронный ресурс]: <https://vue-chartjs.org/guide/#introduction>
22. Get started with BootstrapVue [Электронный ресурс]: <https://bootstrap-vue.org/docs>
23. Wittig A.: Amazon Web Services in Action / A. Wittig, M. Wittig. – К.: «Manning Publications», 2015. – 528 с.
24. AWS documentation [Электронный ресурс]: <https://docs.aws.amazon.com/>
25. PlantUML documentation [Электронный ресурс]: <https://plantuml.com/en/>
26. Terraform Documentation [Электронный ресурс]:
<https://www.terraform.io/docs/index.html>
27. CircleCI documentation [Электронный ресурс]: <https://circleci.com/docs/>

ДОДАТОК А

Приклад інтеграції з системою через HTML код

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Home</title>
  <!-- Copy these lines into header of your website: -->
  <!-- BEGIN -->
  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
  <script src="http://https://wvs-
backend.andlvovsky.com/js/visit.js"></script>
  <script>
    let apiKey = '6a36d434-146b-4457-8d81-d7e6b71bdaf5'; // write the api
key for your site
    visit(apiKey, 'https://wvs-backend.andlvovsky.com', 'v2')
  </script>
  <!-- END -->
</head>
<body>
  <h1>Integration demo</h1>
</body>
</html>
```

ДОДАТОК Б

Програмний скрипт *visit.js* для передачі даних про відвідування

```
function visit(apiKey, url, websiteVersion) {
  $.ajax({
    url: `${url}/visit/${apiKey}`,
    type: "POST",
    data: JSON.stringify({
      time: new Date().toLocaleString(),
      userAgent: navigator.userAgent,
      referralWebsite: new
URL(document.location).searchParams.get("referral"),
      websiteVersion,
      path: document.location.pathname
    }),
    crossDomain: true,
    contentType: "application/json"
  })
}
```

ДОДАТОК В

Конфігураційний файл *config.yml* для налаштування неперервної інтеграції

```
version: 2.1

jobs:
  test_backend:
    machine:
      image: ubuntu-2004:202010-01
      resource_class: medium
      working_directory: ~/backend_repo

    steps:
      - checkout
      - restore_cache:
          keys:
            - gradle-{{ checksum "backend/build.gradle" }}
      - run:
          name: "Run backend tests"
          command: |
            cd backend
            ./gradlew clean test
      - save_cache:
          paths:
            - ~/.gradle
          key: gradle-{{ checksum "backend/build.gradle" }}
```

```
workflows:
  version: 2

  backend:
    jobs:
      - test_backend
```