

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М.Є. Жуковського
«Харківський авіаційний інститут»

Факультет програмної інженерії та бізнесу

Кафедра інженерії програмного забезпечення

Пояснювальна записка до дипломної роботи

магістра
(освітній ступінь)

на тему «Дослідження методів і способів ідентифікації користувачів
на основі їх мережевої активності»

ХАІ.603.667п1.121.156342.200

Виконав: студент 6 курсу групи №667П1
Спеціальність 121 – Інженерія програмного
забезпечення

(код та найменування)

Освітня програма Хмарні обчислення
та Інтернет речей

(найменування)

Лучшева О.П.

(прізвище й ініціали студента)

Керівник: Лучшев П.О.

(прізвище й ініціали)

Рецензент: Іванісенко І. М.

(прізвище й ініціали)

Міністерство світи і науки України
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Факультет програмної інженерії та бізнесу

(повне найменування)

Кафедра інженерії програмного забезпечення

(повне найменування)

Рівень вищої освіти другий (магістерський)

Спеціальність 121 – інженерія програмного забезпечення

(код та найменування)

Освітня програма хмарні обчислення та Інтернет речей

(найменування)

ЗАТВЕРДЖУЮ
Завідувач кафедри

(підпис) (ініціали та прізвище)
“ ” _____ 2020 року

З А В Д А Н Н Я
НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ

Лучшева Олександра Павлівна

(прізвище, ім'я, по батькові)

1. Тема дипломної роботи Дослідження методів і способів ідентифікації користувачів на основі їх мережевої активності

керівник дипломної роботи Лучшев Павло Олександрович, к.т.н., доц. каф 603

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом Університету № _____ від “ ” _____ 2020 року

2. Термін подання студентом роботи 4 грудня 2020 року

3. Вихідні дані до роботи проведений аналіз та порівняння методів та способів ідентифікації користувачів на основі їх мережевої активності, план експерименту, експеримент та оцінка

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)

1 Аналіз існуючих рішень, методів та способів ідентифікації користувачів.

2 Планування експерименту

3 Проведення експерименту та аналіз результатів

5. Перелік графічного матеріалу пояснювальна записка дипломної роботи містить 100 сторінок, 3 додатка, 31 рисунок, 11 таблиць, 55 посилань на літературні джерела; презентація містить 17 слайдів.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	Лучшев П.О., доцент		
2	Лучшев П.О., доцент		
3	Лучшев П.О., доцент		

Нормоконтроль _____
(підпис)

Постернакова В.А. «11» грудня 2020р.
(ініціали та прізвище)

7. Дата видачі завдання «01» вересня 2020 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломної роботи	Строк виконання етапів роботи	Примітка
1	Отримання і затвердження теми диплому	01.09.2020	
2	Обґрунтування актуальності дослідження методів та способів ідентифікації користувачів на основі їх мережевої активності	12.09.2020	
3	Формулювання мети, об'єкту, предмету та методів дослідження	15.09.2020	
4	Аналіз існуючих методів та способів ідентифікації користувачів	20.09.2020	
5	Аналіз існуючих систем	01.10.2020	
6	Аналіз алгоритму та порядку обробки зображення	10.10.2020	
7	Планування експерименту	21.10.2020	
8	Виконання експерименту ідентифікації користувачів у мережі Інтернет	05.11.2020	
9	Підготовка пояснювальної записки	20.11.2020	
10	Оформлення пояснювальної записки до дипломного проекту	22.11.2020	
11	Передзахист дипломної роботи	27.11.2020	
12	Захист дипломної роботи	04.12.2020	

Студент _____ Лучшева О.П.
(підпис) (прізвище та ініціали)

Керівник проекту _____ Лучшев П.О.
(підпис) (прізвище та ініціали)

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

МП – мобільний пристрій;

ОС – операційна система;

ПК – персональний комп'ютер;

ПЗ – програмне забезпечення;

GA – Google Analytics.

РЕФЕРАТ

Пояснювальна записка на дипломну роботу магістра містить 100 сторінок, 31 ілюстрація, 11 таблиць, 3 додатка та 55 джерел.

Метою дипломної роботи є дослідження способів та методів ідентифікації користувачів на основі їх мережевої активності.

Об'єкт дослідження: Процес ідентифікації користувачів у мережі Інтернет.

Предмет дослідження: Методи та способи ідентифікації користувачів у мережі, які базуються на специфічних властивостях програмно-апаратних засобів та поведінки користувачів у мережі.

Методи дослідження: Досягнення мети базуватиметься на основі статистичного аналізу даних, що дозволить проаналізувати отримані результати та зробити висновки щодо того який з методів або у методи у комбінаціях будуть працювати найбільш ефективно.

Наукова новизна отриманих результатів: Отримані дані допоможуть у розвитку безпеки різноманітних бізнес-систем критичного значення та прискорять або полегшать процес ідентифікації користувачів.

Практична значущість роботи: отримані результати допоможуть розробникам програмного забезпечення застосувати додатковий рівень безпеки у бізнес системах критичного призначення.

Ключові слова: КОРИСТУВАЧ, БІЗНЕС-СИСТЕМА КРИТИЧНОГО ПРИЗНАЧЕННЯ, ІДЕНТИФІКАЦІЯ, СТАТИСТИЧНИЙ АНАЛІЗ, ДАНІ, СИСТЕМА, КОРИСТУВАЧ, БЕЗПЕКА.

РЕФЕРАТ

Пояснительная записка на дипломную работу магистра содержит 100 страниц, 31 иллюстрация, 11 таблиц, 3 приложения и 55 источников.

Целью дипломной работы является исследование способов и методов идентификации пользователей на основе их сетевой активности.

Объект исследования: Процесс идентификации пользователей в сети Интернет.

Предмет исследования: Методы и способы идентификации пользователей в сети, которые базируются на специфических свойствах программно-аппаратных средств и поведения пользователей в сети.

Методы исследования: Достижение цели будет базироваться на основе статистического анализа данных, что позволит проанализировать полученные результаты и сделать выводы относительно того, какой из методов или в методы в комбинациях будут работать наиболее эффективно.

Научная новизна полученных результатов: Полученные данные помогут в развитии безопасности различных бизнес-систем критического значения и ускорят или облегчат процесс идентификации пользователей.

Практическая значимость работы: полученные результаты помогут разработчикам программного обеспечения применить дополнительный уровень безопасности в бизнес системах критического назначения.

Ключевые слова: ПОЛЬЗОВАТЕЛЬ, БИЗНЕС-СИСТЕМА КРИТИЧЕСКОГО НАЗНАЧЕНИЯ, ИДЕНТИФИКАЦИЯ, АНАЛИЗ, ДАННЫЕ, СИСТЕМА, ПОЛЬЗОВАТЕЛЬ, БЕЗОПАСНОСТЬ.

ABSTRACT

The explanatory note for the master's thesis contains 100 pages, 31 illustrations, 11 tables, 3 appendices and 55 sources.

The purpose of the thesis is to study ways and methods of identifying users based on their network activity.

Object of research: The process of identifying users on the Internet.

Subject of research: Methods and ways of identifying users in the network, which are based on the specific properties of software and hardware and user behavior in the network.

Research methods: Achieving the goal will be based on statistical analysis of data, which will analyze the results and draw conclusions about which of the methods or methods in combination will work most effectively.

Scientific novelty of the obtained results: The obtained data will help to develop the security of various business systems of critical importance and speed up or facilitate the process of user identification.

Practical significance of the work: the obtained results will help software developers to apply an additional level of security in critical business systems.

Key words: USER, SAFETY-CRITICAL BUSINESS SYSTEM, IDENTIFICATION, ANALYSIS, DATA, SYSTEM, USER, SECURITY.

ЗМІСТ

Вступ.....	10
1 Аналіз способів та методів ідентифікації користувачів.....	12
1.1 Сучасні способи ідентифікації людей.....	12
1.2 Основи інформаційно-аналітичної системи.....	18
1.3 Огляд систем для аналізу та ідентифікації користувачів	19
1.4 Порівняльна таблиця систем ідентифікації кориистувачів	26
1.5 Моделі та методи ідентифікації користувачів	27
1.6 Висновки по розділу 1	38
2 Планування експерименту дослідів параметрів для ідентифікації користувачів у мережі.....	39
2.1 Вибір характеристик	39
2.2 Застосування hash-функцій	43
2.3 Уточнення умов проведення експерименту	44
2.4 Детальний розгляд обраної системи: Google Analytics	44
2.5 Статистичний аналіз	56
2.6 Висновки з розділу 2.....	58
3 Проведення експерименту й аналіз результатів	59
3.1 Проведення експерименту	59
3.2 Отримані дані від Google Analytics	63
3.3 Миття хешування.....	65
3.4 Пояснення отриманих результатів	65
3.5 Аналіз отриманих даних.....	67
3.6 Практичне використання результатів експерименту	69
3.7 Висновки по розділу 3	71
Висновок	72
Перелік посилань.....	73
<i>Додаток А</i> Функція хешування зображення.....	79
<i>Додаток Б</i> Експериментальні дані з Google Analytics	82

<i>Додаток В</i> Експериментальні дані	90
--	----

ВСТУП

Актуальність роботи: Сьогодні у світі майже кожен користується банківськими системами та переймається безпекою власних коштів, а отже підвищення надійності та безпечності критичних бізнес-систем буде завжди актуальною задачею.

Зазвичай система потребує логін та пароль задля верифікації користувача, але крім цього вона збирає й інші дані, такі як ОС пристрою, IP-адресу, час входу до системи та багато іншого, що може бути використано як ключові елементи для ідентифікації користувачів у мережі Інтернет. Прикладом такого застосування є використання поняття довіреного пристрою у деяких сервісах Google.

Мета дослідження: Метою дипломної роботи є дослідження способів та методів ідентифікації користувачів на основі їх мережевої активності.

Задачі дослідження:

- 1 Проаналізувати сучасні методи та способи ідентифікації користувачів, які підключаються з різних пристроїв, браузерів, операційних систем та інше.
- 2 Спланувати експеримент та провести статистичний аналіз факторів, які можуть бути використані для ідентифікації.
- 3 Проаналізувати отримані результати експерименту та надати практичні рекомендації щодо застосування засобів ідентифікації.

Об'єкт дослідження: Процес ідентифікації користувачів у мережі Інтернет.

Предмет дослідження: Методи та способи ідентифікації користувачів у мережі, які базуються на специфічних властивостях програмно-апаратних засобів користувача для роботи у мережі та його поведінки.

Методи дослідження: Досягнення мети базуватиметься на основі статистичного аналізу даних, що дозволить проаналізувати отримані результати та зробити висновки щодо того який з методів або у методи у комбінаціях будуть працювати найбільш ефективно.

Наукова новизна отриманих результатів: Отримані дані допоможуть у розвитку безпеки різноманітних бізнес-систем критичного значення та прискорять або полегшать процес ідентифікації користувачів.

Практична значущість роботи: отримані результати допоможуть розробникам програмного забезпечення для бізнес систем критичного значення.

1 АНАЛІЗ СПОСОБІВ ТА МЕТОДІВ ІДЕНТИФІКАЦІЇ КОРИСТУВАЧІВ

1.1 Сучасні способи ідентифікації людей

Останнім часом значно підвищився інтерес до тематики цифрової ідентифікації особистості, що прийнято пов'язувати із зростанням загроз міжнародного тероризму, проблемами процесів демократизації при переході до інформаційного суспільства. Інтерес до цифрової ідентифікації особи зростає також у зв'язку із збільшенням обсягів торговельних операцій, здійснюваних через глобальні комп'ютерні мережі, зокрема через Інтернет. Необхідність регламентації доступу виникла з незапам'ятних часів і пов'язана з природним бажанням кожної людини захистити те, чим він володіє. Але регламентація доступу неможлива без ідентифікації особи за принципом "свій - чужий".[10]

Сьогодні існує декілька способів ідентифікації користувачів. У кожного з них є свої переваги і недоліки, завдяки чому деякі технології підходять для використання в одних системах, інші – в інших. Однак у багатьох випадках немає строго певного рішення. А тому як розробникам програмного забезпечення, так і користувачам приходиться самостійно думати, який спосіб ідентифікації реалізовувати в продуктах.

1.1.1 Парольна ідентифікація

Ще не дуже давно парольна ідентифікація була чи ледве не єдиним способом визначення особистості користувача. І в цьому немає абсолютно нічого дивного. Справа в тому, що парольна ідентифікація найбільш проста як у реалізації, так й у використанні. Суть її зводиться до наступного. Кожен зареєстрований користувач якої-небудь системи одержує набір персональних реквізитів (звичайно використовуються пари логин-пароль). Далі при кожній спробі входу людина повинна вказати свою інформацію. Ну а оскільки вона

унікальна для кожного користувача, то на підставі її система й робить висновок про особистість та ідентифікує.

Головна перевага парольної ідентифікації вже вказана - це простота реалізації й використання. Крім того, введення парольної ідентифікації не вимагає зовсім ніяких витрат: даний процес реалізований у всіх програмних продуктах, що є в продажу. Таким чином, система захисту інформації виявляється гранично простою і доступною.

Тепер перейдемо до недоліків. На жаль, їх багато. І самий, мабуть, головний – величезна залежність надійності ідентифікації і від самих користувачів, точніше, від обраних ними паролів. Справа в тому, що більшість людей використовують ненадійні ключові слова, які легко підбираються. До них ставляться занадто короткі паролі, що мають зміст слова й т.д. Тому деякі фахівці в області інформаційної безпеки радять використати довгі паролі, що складаються з безладного сполучення букв, цифр і різних символів. От тільки користувачі не хочуть запам'ятовувати такі ключові слова й починають записувати їх на папері, які приклеюють прямо до монітора або ховають під клавіатуру. Чи потрібно говорити, що подібні дії – серйозний удар по інформаційній безпеці.

1.1.2 Апаратна ідентифікація

Розглянутий принцип ідентифікації ґрунтується на визначенні особистості користувача по якомусь предметі, ключу, що перебуває в його ексклюзивному користуванні. Природно, мова йде не про звичайних, звичних для більшості людей ключах, а про спеціальні електронні. На даний момент найбільше поширення одержали два типи пристроїв.

До першого ставляться всілякі карти (рисунок 1.1). Їх досить багато, і працюють вони по різних принципах. Так, наприклад, досить зручні у використанні безконтактні карти (їх ще називають проксіміті-карти), які дозволяють користувачам проходити ідентифікацію як у комп'ютерних

системах, так й у системах доступу в приміщення. Найбільш надійними вважаються смарт-карти - аналоги звичних багатьом людям банківських карт. Крім того, є й більш дешеві, але менш стійкі до злому карти: магнітні, зі штрих-кодом і т.д.



Рисунок 1.1 – Ключ-картка

Іншим типом ключів, які можуть використатися для апаратної ідентифікації, є так називані токени. Ці пристрої мають власну захищену пам'ять і підключаються безпосередньо до одного з портів комп'ютера (USB, LPT) (рисунок 1.2).



Рисунок 1.2 – USB пристрій

Головною перевагою застосування апаратної ідентифікації є досить висока надійність. І дійсно, у пам'яті токенів можуть зберігатися ключі, підібрати які хакерам не вдасться. Крім того, у них реалізовано чимало різних захисних механізмів. Ну а вбудований мікропроцесор дозволяє електронному ключу не тільки брати участь у процесі ідентифікації користувача, але й виконувати деякі

інші корисні функції. Мабуть, найбільш серйозною небезпекою у випадку використання апаратної ідентифікації є можливість крадіжки зловмисниками токенів у зареєстрованих користувачів. Втім, ця проблема легко вирішується за допомогою застосування багатофакторної ідентифікації (про те, що це таке, буде сказано нижче).

Тепер про недоліки апаратної ідентифікації. Про одному з них вже згадали - це можливість крадіжки електронних ключів. Другий мінус розглянутої технології - ціна. Взагалі ж останнім часом вартість як самих електронних ключів, так і програмного забезпечення, що може працювати з ними, помітно знизилася. Проте, для введення в експлуатацію системи майнової ідентифікації однаково будуть потрібні деякі вкладення. Все-таки кожного зареєстрованого користувача (або хоча б привілейованих користувачів - адміністраторів, керівництво підприємства й т.д.) потрібно забезпечити персональними токенами. Крім того, згодом деякі типи ключів можуть зношуватися, крім того, вони можуть бути загублені й т.д. Тобто майнова ідентифікація вимагає деяких експлуатаційних витрат.

1.1.3 Біометрична ідентифікація

Біометрія - це ідентифікація людини по унікальним, властивим тільки ньому біологічним ознакам. Тобто, можна сказати, що біометричні технології споконвічно розроблялися для точного встановлення особистості людини. А тому рішення використати їх в області інформаційної безпеки виглядає цілком логічним. Причому даний напрямок розвивається дуже активно. Сьогодні експлуатується вже більше десятка різних біометричних ознак. Причому для найпоширеніших з них (відбитки пальців і райдужна оболонка ока) існує безліч різних за принципом дії сканерів. Так що користувачам, що вирішили використати біометричну ідентифікацію, є із чого вибрати.

Головною перевагою біометричних технологій є найвища надійність. І дійсно, усі знають, що двох людей з однаковими відбитками пальців у природі

просто не існує. Правда, сьогодні вже відомо кілька способів обману дактилоскопічних сканерів. Наприклад, потрібні відбитки пальців можуть бути перенесені на плівку або до пристрою може бути прикладена велика фотографія пальця зареєстрованого користувача. Втім, треба зізнатися, що сучасні пристрої вже не попадаються на такі прості виверти. Так що зловмисникам доводиться видумувати всі нові й нові способи обману біометричних сканерів, багато хто з яких вимагають роботи висококласних фахівців і дуже дорогого встаткування.



Рисунок 1.3 – Біометрична ідентифікація

Основним недоліком біометричної ідентифікації є вартість устаткування. Адже для кожного комп'ютера, що входять до цієї системи, необхідно придбати власний сканер. Звичайно, останнім часом ціни на біометричні пристрої постійно падають. Крім того, не дуже давно з'явилися миші й клавіатури з убудованими дактилоскопічними сканерами. Причому їхня ціна ненабагато відрізняється від вартості "звичайної" периферії. Правда, варто відзначити, що подібні дешеві сканери недовговічні. Крім того, у них досить високий відсоток помилок другого роду (відмова в доступі зареєстрованому користувачеві). Тому користувачеві доводиться вибирати, яке пристрій придбати - по дорожче й краще або подешевше й гірше.

1.1.4 Багатофакторна ідентифікація

До дійсного моменту вели мову про однофакторну ідентифікацію. Тобто в розглянутих системах для визначення особистості користувача використовувався тільки один фактор. Однак подібні процеси сьогодні не можна назвати надійними. Наприклад, зловмисник може вкрасти токен у зареєстрованого користувача й легко скористатися ним для несанкціонованого доступу до інформації. Саме тому поступово все більшого поширення одержує багатофакторна ідентифікація, коли для визначення особистості застосовується відразу кілька параметрів.



Рисунок 1.4 – Пристрій для багатофакторної ідентифікації

Причому комбінуватися ці фактори можуть у довільному порядку. Втім, сьогодні в переважній більшості випадків використовується тільки одна пара: парольний захист і токен. У цьому випадку користувач може не боятися підбора його пароля зловмисником (без електронного ключа вона працювати не буде), а також крадіжки токена (він не буде працювати без пароля). Втім, у деяких системах застосовуються максимально надійні, можна навіть сказати перебільшено надійні процедури ідентифікації. У них одночасно використовуються паролі, токени й біометричні характеристики людини.[11]

1.2 Основи інформаційно-аналітичної системи

Інформаційно-аналітичні системи визначаються як особливий клас інформаційних систем, призначених для аналітичної обробки даних, а не для автоматизації повсякденної діяльності організації. Інформаційно-аналітичні системи об'єднують, аналізують і зберігають як єдине ціле інформацію, яка вилучена як з баз даних організації, так і із зовнішніх джерел. Сховища даних, що входять до складу інформаційно-аналітичних систем, забезпечують перетворення великих об'ємів деталізованих даних в узагальнену вивірену інформацію, яка придатна для прийняття обґрунтованих рішень. На відміну від звичайних баз даних сховища містять оброблене, впорядковане і зрозуміле керівникам представлення даних.; вони стають складальним конвеєром по підготовці інформації в інтегрованому, несуперечливому, наочному вигляді для підтримки прийняття управлінських рішень.

Створення інформаційно-аналітичних систем, що реально відповідають цілям і завданням організацій, визнається як достатньо складний процес, що включає етапи формування концепції, проектування, розробки, впровадження і супроводу. Сам характер цього процесу вимагає попередньої розробки фіксованої технологічної схеми. Технологічна схема відповідає стандарту, що описує процеси життєвого циклу програмних засобів, послідовність робіт і завдань, що виконуються певними виконавцями. Таким чином, необхідна загальна методика створення інформаційно-аналітичних систем, що містить склад і послідовність робіт і завдань, склад ролевих функцій і породжуваних артефактів (документів, моделей, схем і ін.).

Технологія і методика створення інформаційно-аналітичних систем охоплює наступні види діяльності:

- збір, аналіз і деталізацію вимог до інформаційно-аналітичної системи, визначення пріоритетів реалізації цих вимог і постановка завдань по їх реалізації, визначення вимог по архітектурі, надійності і захисту від несанкціонованого доступу і визначення складу даних;

- розробка проектних рішень по всіх аспектах побудови інформаційно-аналітичної системи, визначення складу джерел інформації, способів передачі даних, складу додатків організації доступу до даних, проектування архітектури, проектування баз даних;
- розробка аналітичних додатків, вибір і настройка інструментальних засобів збору, перетворення і очищення даних і організації доступу користувачів до даним, розробка метаданих, тестування, розробка документації для користувачів[12].

1.3 Огляд систем для аналізу та ідентифікації користувачів

Наразі уснує багато різних систем для аналізу та/або ідентифікації користувачів інших систем або сайтів. У роботі розглянемо 5 популярних систем та проаналізуємо їх, щоб обрати кращу для подальшої роботи.

1.3.1 Огляд системи Google Analytics

Google Analytics - це безкоштовний сервіс для збору інформативної статистики по відвідувачам сайтів. Дані збираються за допомогою коду на сторінках, він спрацьовує після відкриття сторінки користувачем.[1]

Переваги та недоліки наведені у таблиці 1.1, а приклад роботи на рисунку 1.5.

Таблиця 1.1 – Переваги та недоліки Google Analytics

Переваги	Недоліки
Платформи: web, Android, iOS	Відсутня багатofакторна авторизація
Розгортання: хмара	Відсутнє резервне копіювання у декількох місцях
Велика кількість мов, що підтримуються	Відсутній таргетинг
Безкоштовно	Відсутнє API
Доступ за протоколом HTTPS	

Закінчення таблиці 1.1

Переваги	Недоліки
Відстеження джерел Відстеження ключових слів Відстеження навігації Відстеження конверсій[2] А/В-тестування Відстеження часу перебування Сегментація аудиторії Відстеження популярності посилань	



Рисунок 1.5 – Робота Google Analytics

1.3.2 Огляд системи Amplitude

Amplitude - платформа аналітики, яка покликана показати «поведінковий шар» призначених для користувача даних, це проміжний рівень між вихідними

даними і дашбордами. Amplitude пропонує збільшити масштаб окремих точок для перегляду даних про поведінку користувачів і діях.

Потоки показують, які дії частіше роблять користувачі після підписки, і до покупки. Amplitude дозволяє вимірювати коефіцієнти конверсії за допомогою воронки для ідентифікації точок, в яких користувачі кидають реєстрацію, поновлення або покупку. Можна вимірювати утримання користувача і частоту взаємодії з додатком. Користувачів можна поділити на сегменти і групи для порівняння за різними показниками.[3]

Переваги та недоліки наведені у таблиці 1.2, а приклад роботи на рисунку 1.6.

Таблиця 1.2 – Переваги та недоліки Amplitude

Переваги	Недоліки
Платформа: web	Неповністю безкоштовно
Розгортання: хмара	Лише підтримка англійської мови
Доступ за протоколом HTTPS	Відсутня багатофакторна авторизація
Відстеження джерел	Відсутнє резервне копіювання у декількох місцях
Відстеження навігації	Відсутнє відстеження ключових слів
Відстеження конверсій	Відсутній таргетинг
А/В-тестування	Відсутнє відстеження популярності посилань
Відстеження часу перебування	
Сегментація аудиторії	
API	

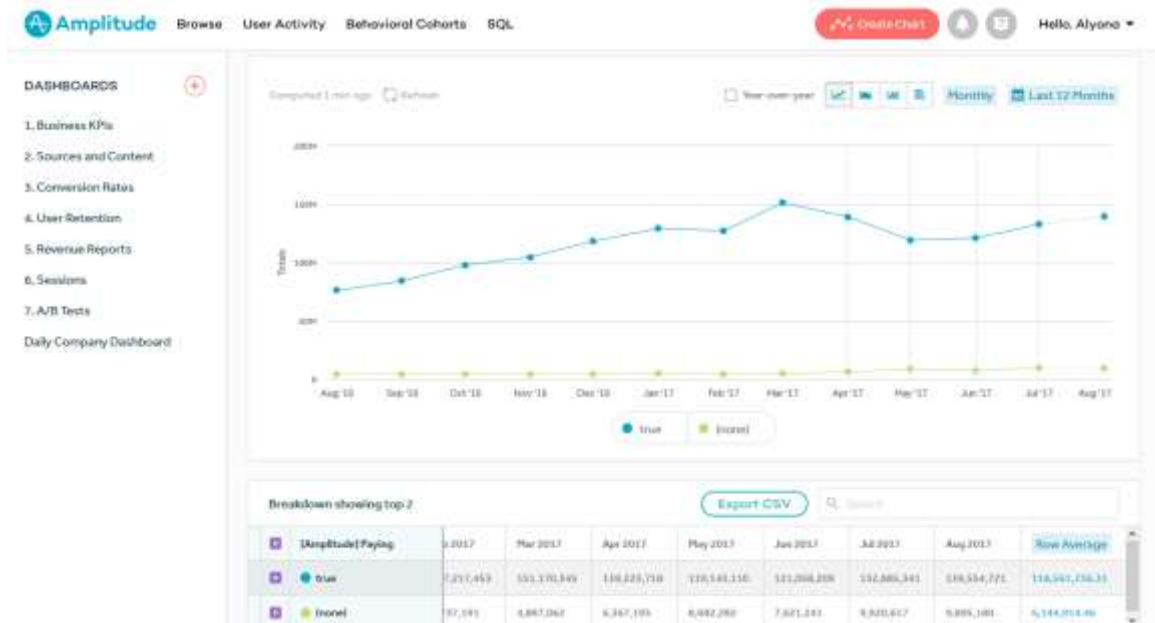


Рисунок 1.6 – Робота Amplitude

1.3.3 Огляд системи IO technologies

IO technologies (раніше Onthe.io) - це онлайн-система аналітики для медіа-проектів. Стежить за поведінкою читачів, оцінює ROI, маркетингові кампанії, різні метрики і надає детальні звіти.

Допомагає маркетологам з вимірюванням рентабельності кампаній, команді продажів - збільшити коефіцієнт конверсії, продуктовим менеджерам - аналізувати послідовність продажів, виявляти проблеми, аналітикам - контролювати події та поведінку відвідувачів.[4]

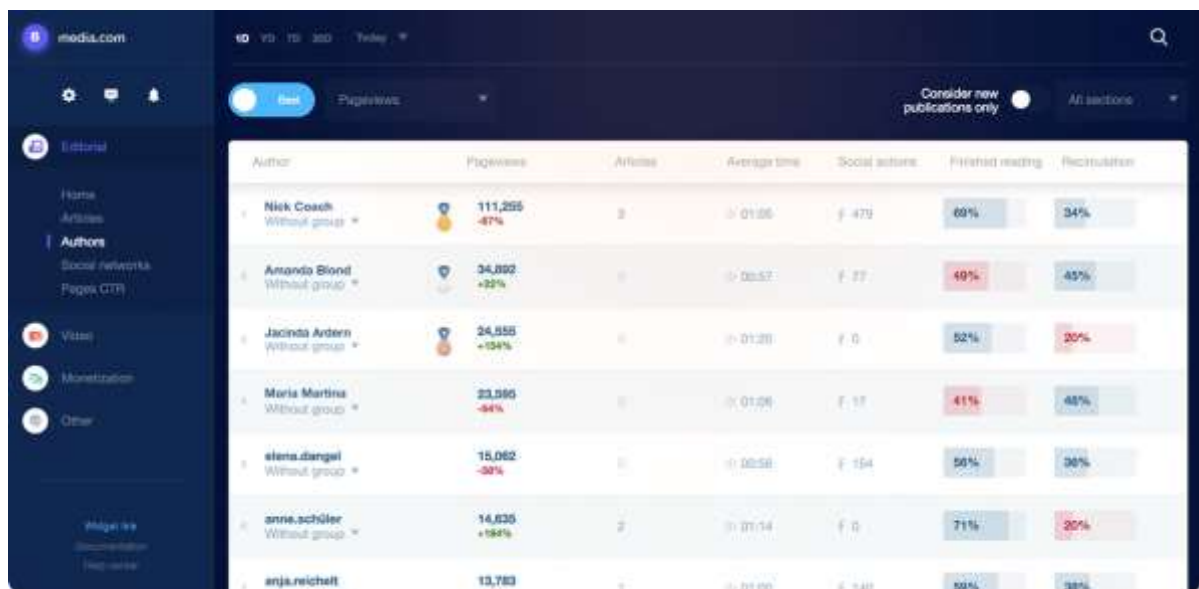
Переваги та недоліки наведені у таблиці 1.3, а приклад роботи на рисунку 1.7.

Таблиця 1.3 – Переваги та недоліки IO technologies

Переваги	Недоліки
Платформа: web	Неповністю безкоштовно
Розгортання: хмара	Лише підтримка англійської мови
Доступ за протоколом HTTPS	Відсутня багатофакторна авторизація

Закінчення таблиці 1.3

Переваги	Недоліки
<p>Відстеження джерел</p> <p>Відстеження навігації</p> <p>Відстеження конверсій</p> <p>A/B-тестування</p> <p>Відстеження часу перебування</p>	<p>Відсутнє резервне копіювання у декількох місяцях</p> <p>Відсутнє відстеження ключових слів</p> <p>Відсутня сегментація аудиторії</p> <p>Відсутній таргетинг</p> <p>Відсутнє API</p> <p>Відсутнє відстеження популярності посилань</p>



Author	Pageviews	Articles	Average time	Social actions	Finished reading	Recruitment
Niek Coeck Without group	111,255 -47%	2	01:06	470	60%	34%
Amanda Biondi Without group	34,002 +25%	0	00:57	27	49%	45%
Jacinda Arden Without group	34,555 +104%	0	01:20	0	52%	20%
Maria Martins Without group	23,590 -44%	0	01:06	17	41%	48%
elena.dargel Without group	15,062 -30%	0	00:58	104	56%	36%
anne.schüler Without group	14,835 +184%	2	01:14	0	71%	20%
anja.reichelt	13,780	1	01:00	140	59%	38%

1.3.4 Огляд системи Segment

Segment - це веб-платформа і API для веб-аналітики та збору даних про користувачів для подальшої їх відправки на сотні інструментів або сховищ даних.

Segment допоможе розібратися в даних про клієнтів, що надходять з різних джерел. Додаток інтегрується з сотнями інших сервісів, обробляючи їх дані в

єдину платформу. Він також отримує дані з веб-сайтів, мобільних пристроїв, браузерів, годин, iBeacons, пунктів продажу та ін.

За допомогою Segment можна експортувати дані в будь-яку внутрішню систему і додаток, відтворювати історичні дані, переглядати події в режимі реального часу, наприклад, коли хтось робить покупку на сайті або в додатку [5].

Переваги та недоліки наведені у таблиці 1.4, а приклад роботи на рисунку 1.8.

Рисунок 1.7 – Робота IO technologies

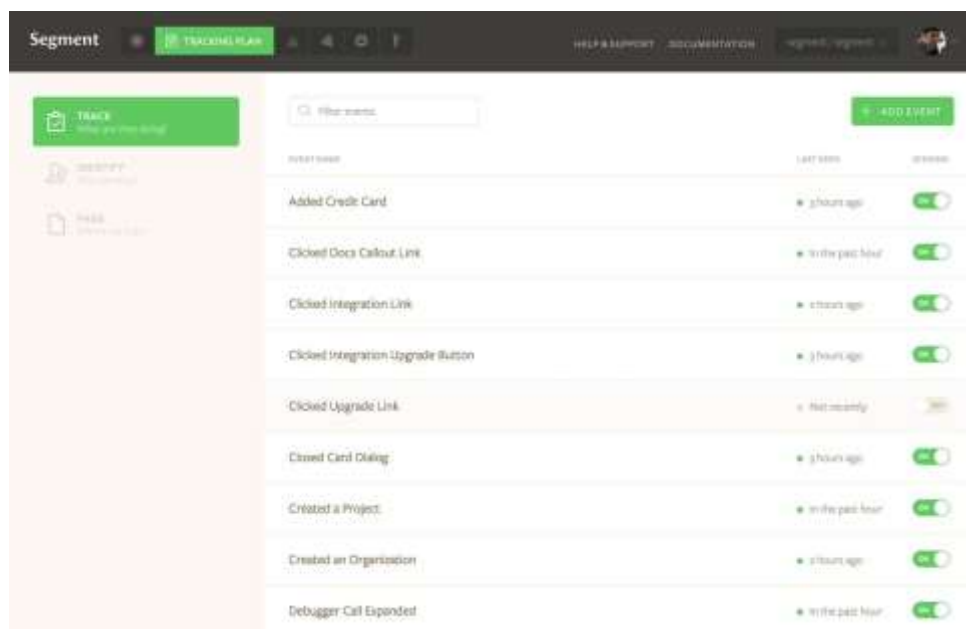


Рисунок 1.8 – Робота Segment

Таблиця 1.4 – Переваги та недоліки Segment

Переваги	Недоліки
Платформа: web	Неповністю безкоштовно
Розгортання: хмара	Лише підтримка англійської мови
Доступ за протоколом HTTPS	Відсутня багатофакторна авторизація
Відстеження джерел	Відсутнє резервне копіювання у декількох місцях
Відстеження конверсій	

Закінчення таблиці 1.4

Переваги	Недоліки
А/В-тестування	Відсутнє відстеження навігації
Відстеження часу перебування	Відсутнє відстеження ключових слів
Сегментація аудиторії	Відсутнє А/В-тестування
Таргетинг	Відсутнє відстеження популярності
API	посилань

1.3.5 Огляд системи Chartbeat

Chartbeat – це невелика онлайн-система для аналізу контенту. Вона орієнтована на редакторів, письменників, розробників і всіх тих, хто публікує контент і хоче повноцінної аналітики по взаємодії з ним користувачів [6].

Переваги та недоліки наведені у таблиці 1.5, а приклад роботи на рисунку 1.5.

Таблиця 1.5 – Переваги та недоліки Chartbeat

Переваги	Недоліки
Платформа: web	Неповністю безкоштовно
Розгортання: хмара	Лише підтримка англійської мови
Відстеження джерел	Відсутній доступ за протоколом HTTPS
Відстеження навігції	Відсутня багатофакторна авторизація
Відстеження часу перебування	Відсутнє резервне копіювання у декількох місцях
	Відсутнє відстеження конверсій
	Відсутнє відстеження ключових слів
	Відсутнє А/В-тестування
	Відсутнє відстеження ключових слів
	Відсутня сегментація аудиторії
	Відсутній таргетинг

Закінчення таблиці 1.5

Переваги	Недоліки
	Відсутній таргетинг Відсутнє API Відсутнє відстеження популярності посилань

1.4 Порівняльна таблиця систем ідентифікації користувачів

У таблиці 1.6 наведено порівняльні характеристики оглянутих систем, за якими обрано підходящу систему.

Таблиця 1.6 – Порівняльні характеристики систем

Характеристики/назва продукту	Google analytics	Amplitude	IO technologies	Segment	Chartbeat
Web	+	+	+	+	+
Android	+				
iOS	+				
Хмарний сервіс	+	+	+	+	+
Підтримка більше 3 мов	+				
Безкоштовно	+				
Доступ за HTTPS	+	+	+	+	
Багатофакторна авторизація					
Резевне копіювання у декількох місцях					
Відстеження джерел	+	+	+	+	+
Вістеження ключових слів	+				
Відстеження навігації	+	+	+		+
Відстеження конверсій	+	+	+	+	
a/b-тестування	+	+	+	+	
Відстеження часу перебування	+	+	+	+	+

Закінчення таблиці 1.6

Характеристики/назва продукту	Google analytics	Amplitud e	IO technologies	Segmen t	Chartbea t
Сегментація аудиторії	+	+		+	
Таргетинг				+	
Відстеження популярності посилань	+				
API		+		+	

Розглянувши різноманітні системи та їх порівняння із Google Analytics, було вирішено використовувати у подальшій роботі Google Analytics.

1.5 Моделі та методи ідентифікації користувачів

Веб-відстеження – це процес, за допомогою якого сторони, які переглядають веб-трафік, ідентифікують характерні схеми навігації, щоб віднести історію перегляду до певних осіб. Сторонні трекери залишаються основною проблемою; їх поширеність і активність масового відстеження добре задокументовані [42, 43, 44].

На початку 2020 року світ перебуває в умовах потрясіння екосистеми відстеження, оскільки регуляторний нагляд та публічний дискурс, як видається, досягли найвищої точки. Робота, опублікована за часів оригінального рукопису, дає уявлення про стан відстежувальної екосистеми тієї епохи [39,41,45,46]. Пізніші роботи зображують все більш досконалі технології відстеження, що підживлюють цільову галузь поведінкової реклами [47, 50, 51]. Також спостерігається постійне збільшення масштабу [52], глибоку відсутність прозорості у розкритті потоків особистої інформації [34] та консолідацію Інтернет-економіки для меншої кількості більших, домінуючих партій [40]. Концепція єдиної мережі стає застарілою, оскільки все більше і більше вмісту

динамічно генерується, персоналізується на кожного відвідувача та генерується самими відвідувачами Інтернету.

1.5.1 Методологія OpenWPM

Розроблена методологія дозволить не лише відтворити оригінальні висновки, але й розширити аналіз до конкретних загроз конфіденційності, викликаних роботами оригінальних авторів. Дані були зібрані від ~ 52 000 користувачів браузера Firefox, які вирішили обмінюватися даними для досліджень та розробки продуктів, окрім того, що зазначено в політиці збору даних за замовчуванням Mozilla [35]. Учасникам-кандидатам було показано відповідне запитання згідно з політикою Mozilla, яка регулює управління даними та управління згодами на дослідження користувачів [38]. Ця підказка надала користувачам чітке, зрозуміле англійською мовою пояснення розширеного збору даних [36].

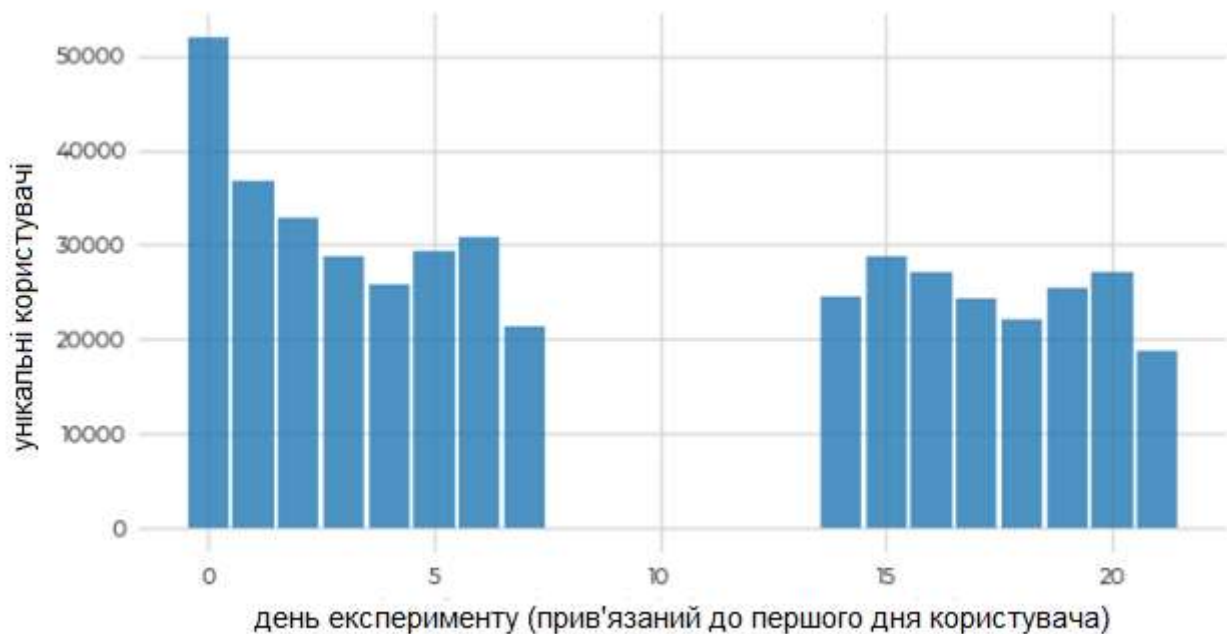
Вимірювання даних перегляду здійснювалось за допомогою власного розширення браузера [37], отриманого з інструментарію OpenWPM [39]. Дані були зашифровані на клієнті та зібрані за допомогою захищеної інфраструктури, відокремленої від звичайного телеметричного конвеєра Mozilla, що забезпечує дуже обмежений доступ до даних. Дані збирались у ring, які регулярно передавались із браузера у міру накопичення даних. Кожному браузеру був наданий унікальний ідентифікатор, щоб дозволити об'єднати ring для збору набору даних. Цей унікальний ідентифікатор був специфічним для програми збору даних, яка використовується в цьому дослідженні, і не підключена до інших ідентифікаторів, що використовуються Mozilla.

Для кожного користувача необхідно було накопичити два різні періоди перегляду даних. Практичні міркування, що стосуються загального обсягу зібраних даних, експлуатаційних витрат та бажання виміряти стабільність профілю, спонукали реалізацію, яка збирала дані протягом 7 календарних днів,

робила паузу на 7 днів і згодом поновлювалась на додаткові 7 днів. Було додано додатковий день до кожного тижневого періоду спостереження, щоб забезпечити збір пінгів із затримкою. підсумку містив дані веб-перегляду за ~35 мільйонів відвідувань сайтів та ~660 000 різних доменів, зібраних між 16 липня та 13 серпня 2019 року. На рисунку 1.14 показано розподіл кількості різних доменів на одного користувача, сукупного за день збору. Зазвичай видно медіану 8 різних доменів на користувача на день; сукупність за весь період збору призводить до медіани 34 різних доменів на користувача.

Обмежуючи кількість доменів лише до списку найпопулярніших веб-сайтів, середня кількість доменів на клієнта становила 18. Це порівняно з медіаною 10, яку можна побачити в оригінальній статті, методика збору даних якої вимагала використання список топ-сайтів. Також зауважено, що максимальна кількість доменів на одного користувача – 1116.

На рисунку 1.14 показано кількість унікальних користувачів, які були активними кожен день експерименту.



Рисунк 1.9 – Кількість унікальних активних користувачів на день протягом експериментального періоду (включаючи навмисний тижневий розрив)

Хоча це багато, це не нереальна сума за 14 днів перегляду та є одним із показників того, що користувачі експерименту були справжніми користувачами, а не ботами чи іншими автоматизованими інструментами, що використовують Firefox.

Було розраховано розподіл розміру профілів між користувачами. У таблиці 1.9 представлений вищий рівень відносних розмірів профілів при розгляді всіх відвідуваних унікальних доменів. Спостерігається подібна експоненціальна тенденція, як у оригінальній роботі: найбільша кількість користувачів має невеликий розмір історії.

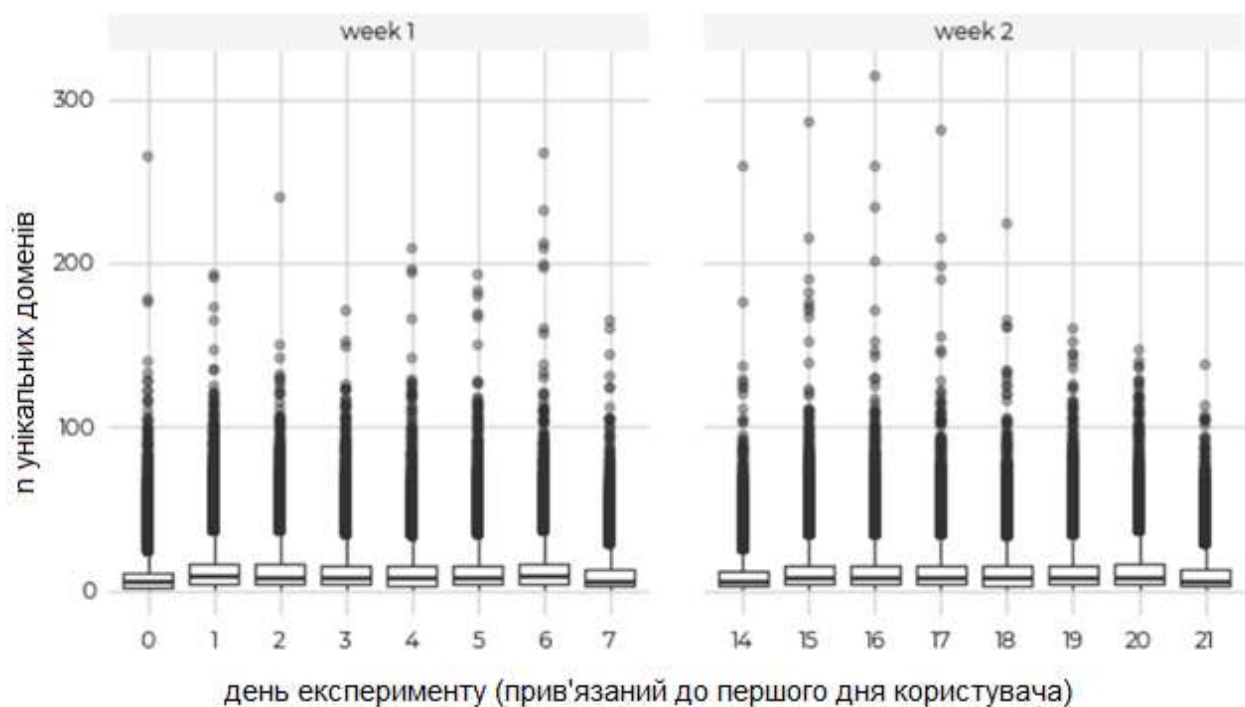


Рисунок 1.10 – Кількість унікальних відвідуваних доменів на одного користувача за день

Дані про кількість відвідувань сайту за день не представляються. Було виявлено, що деякі користувачі були сильно перевищували активність із сотнями тисяч подій завантаження сторінки. Їх було розглянуто вручну і виявлено, що ці користувачі мали один або два сайти, які потрапляли дуже багато разів із звичайною історією браузера. Припускається, що у цих користувачів були встановлені доповнення (браузери) для автоматизації певних

робочих процесів, але в іншому випадку вони були репрезентативними для звичайного перегляду. Оскільки представлено історію перегляду як логічний вектор різних доменів, тим самим видаляючи всі вимірювання частоти, ця подія не вивчалась далі і не видалялись вручну користувачів із набору даних.

Таблиця 1.7 – Кількість користувачів за кількістю унікальних відвідувань домену

Min size	Max size	N users
1	25	21,519
26	50	11,195
51	75	6,750
76	100	4,499
101	125	2,791
126	150	1,766
151	–	3,457
	Total	51,977

Розглянутий механізм збору даних дозволив фіксувати всі мережеві запити та відповіді, пов'язані з подіями навігації на сайті. Починаючи з повної інформації про історію перегляду користувача, можна вивчити унікальність перегляду профілів у повному обсязі та висловити тверді твердження щодо стабільності профілів та моделювати конкретні сценарії перевизначення. Охоплення всіх мережевих запитів дозволяє оцінити потенційні можливості відстеження для широкого кола третіх сторін.

Окрім вищезазначених методологічних відмінностей, відзначаються кілька джерел потенційних змін між конкретними популяціями досліджень. В обох дослідженнях використовується вибір населення, хоча з різною методологією та потоками. Досліджувана когорта дала апіорну згоду на поздовжній збір даних, що означає низький рівень стирання порівняно з оригінальною роботою, яка спиралася на відвідувачів, які повернулись на веб-сайт дослідження. Оригінальне дослідження могло вивчити трафік від користувачів різних веб-браузерів. Вони не повідомляли про відмінності між браузерами. Однак вони

згадують дані від 1256 користувачів, що повідомляють про мобільного агента користувача. WebExtension надійшов до версій Firefox 67 або 68, які були версіями каналу випуску протягом досліджуваного періоду, і забезпечив, щоб браузер учасників був налаштований на локалізацію в США, оскільки текст згоди на участь був доступний лише англійською мовою.

1.5.2 Вибір вектора історії OpenWPM

Оригінальне дослідження Олейника та співавторів потребувало попередньо вибраного списку сайтів, що використовуються для перевірки історії користувачів, щоб дізнатись, чи відвідував їх користувач. Вони відсортували список сайтів за спостережуваною популярністю в їх наборі даних. Тоді впорядкований у всьому світі вектор із логічними записами для кожного користувача був основним вкладом у їх аналіз.

Оскільки наявна повна інформація про історію перегляду протягом досліджуваного періоду, можна створити вектор історії спостережуваних доменів, рангований за популярністю. У результатах це позначено як усі спостережувані домени. Однак також створили заздалегідь визначений список сайтів для аналізу. Це зроблено з наступних причин:

- використання заздалегідь визначеного списку дозволяє більш точно повторити методологію оригінального документу;
- для побудови вектора категорії використовується сторонній сервіс (див. розділ 2.1.2), і без передачі даних користувача до служби аналізу, базуючи наші запити на даних користувачів;
- оскільки атаки, що аналізують історію та трафік та які вимагають заздалегідь визначеного вектора, все ще можливі [54], доцільно проводити аналізи з одним.

Список 6000 популярних сайтів Олейника та інших "був створений із 500 найпопулярніших посилань від Alexa, 4000 зі списку популярних веб-сайтів Quantcast, списків загальнодержавних та військових веб-сайтів та кількох

вибраних на замовлення URL-адрес, вибраних для демонстрації та освітнього потенціалу".

Проведений Zeber [48] перед експериментом аналіз, розглянув різні найкращі списки веб-сайтів і привів до нашого списку сайтів, який є гібридом списків Alexa 10 (10) та Tranco [33]. Назвемо його списком Треха, і він складається шляхом чергування веб-сайтів з кожного та видалення повторюваних записів [53]. 100 найкращих веб-сайтів Треха збігаються із 100 найпопулярнішими сайтами в наших даних користувачів на 40%. Якщо розглянути 10 000 найпопулярніших сайтів із кожного списку, перекриття зросте до 52%.

1.5.3 Генерація вектора категорії OpenWPM

Для створення необхідного вектора категорії використовували API WebShrinker [49] для отримання категоризації доменів. Весь список Треха запускався через API WebShrinker з поверненою категоризацією Бюро інтерактивної реклами. Таксономія IAB має ряд категорій верхнього та підрівневого рівнів з відповідними балами та рівнями довіри для кожної пари домен-категорій. Зіставили кожен домен з найвищою рейтинговою підкатегорією, якщо такої впевненої категорії не існувало. У цьому випадку було замінено найбільш релевантну категорію вищого рівня. Якщо домен було внесено до списку "Без категорії", його видаляли із набору даних. Нарешті, відсортували категорію від найбільш до найменш спостережуваних, як це було зроблено з вектором усіх спостережуваних доменів. Зрештою, категоричний набір даних містив 281 категорію із загальної кількості можливих 404, запропонованих у стандарті категорій IAB. Оригінальний документ використовував подібну службу категоризації під назвою Trend Micro, яка дала 72 категорії інтересів. Хоча Trend Micro існує і сьогодні, обрано WebShrinker, оскільки категоризація IAB відповідає загрози adtech, у якій зацікавлені в дослідженні.

1.5.4 Термінологія OpenWPM

Протягом усієї роботи використовується слово користувач для зручності, але насправді це вивчення наборів історій браузера. Дані, які були зібрані, гарантують лише унікальний ідентифікатор браузера. На одного користувача може бути кілька браузерів, а на браузер може бути кілька користувачів.

Колекції унікальних відвідувань домену користувача (або категорії) за певний проміжок часу як профіль, називають базовим набором даних профілів може бути список усіх спостережуваних відвідувань домену, відвідувань доменів Tреха або категорій. Таким чином, розмір профілю – це кількість унікальних доменів або категорій, які містить профіль.

Як і в оригінальній роботі, також використовується концепція булевого вектора профілю. Індекс цього вектора називається рангом; таким чином, підвектор-до-рангу- k описує вектор довжини k з логічними значеннями, що представляють, чи відвідував користувач кожен з найпопулярніших k елементів у наборі даних. Нагадаємо, що елементи у векторі відсортовані від найпопулярніших елементів у базовому наборі даних до найменш популярних.

1.5.5 Унікальність профілю веб-історії OpenWPM

На рисунку 3.1 порівнюються розмір профілів, у спостереженні всіх доменів (як показано в таблиці 2.1), коли використовується список Tреха для вказівки набору доменів, що розглядаються, і коли використовується представлення категорій профілів. Як і слід було очікувати, зміни розподілу залишились, оскільки профілі зменшувались у розмірах відносно рівномірно для всієї сукупності. Переходячи з усіх доменів до списку Tреха, медіанний розмір профілю зменшується до 18 доменів. Обмежуючись лише категоріями, медіанний розмір профілю все ще змінюється, до 11. Зазначено, що хоча максимальний розмір профілю для всіх спостережуваних доменів (1116)

набагато вищий, ніж аналог для Треха (288) та категорій (73), спостерігається понад 99% усіх користувачів у цьому сюжеті.

На рисунку 1.12 розглянуто розподіл поширеності профілів історії, побудованих із набору всіх спостережуваних доменів, доменів Треха та категорій сторінок. Крайня ліва точка на осі x представляє найпоширеніший профіль для кожного базового набору даних. Вісь y представляє кількість користувачів з цим профілем. Наприклад, у наборі даних Треха найпопулярніший профіль складається лише з відвідувань найпопулярнішого домену Треха, www.google.com, профілю, яким користуються 559 користувачів при розгляді всіх доменів у списку Треха (коричнева лінія) і майже 10 000 користувачів, коли домен, встановлений для створення профілів, обмежений лише 10 найкращими сайтами (синя лінія). Якщо знову розглянути всі домени у списку Треха, другий за популярністю профіль складається лише з відвідувань www.youtube.com із 150 користувачами. Як правило, чим менший пул користувачів з однаковим профілем, тим легше можна визначити конкретного користувача.

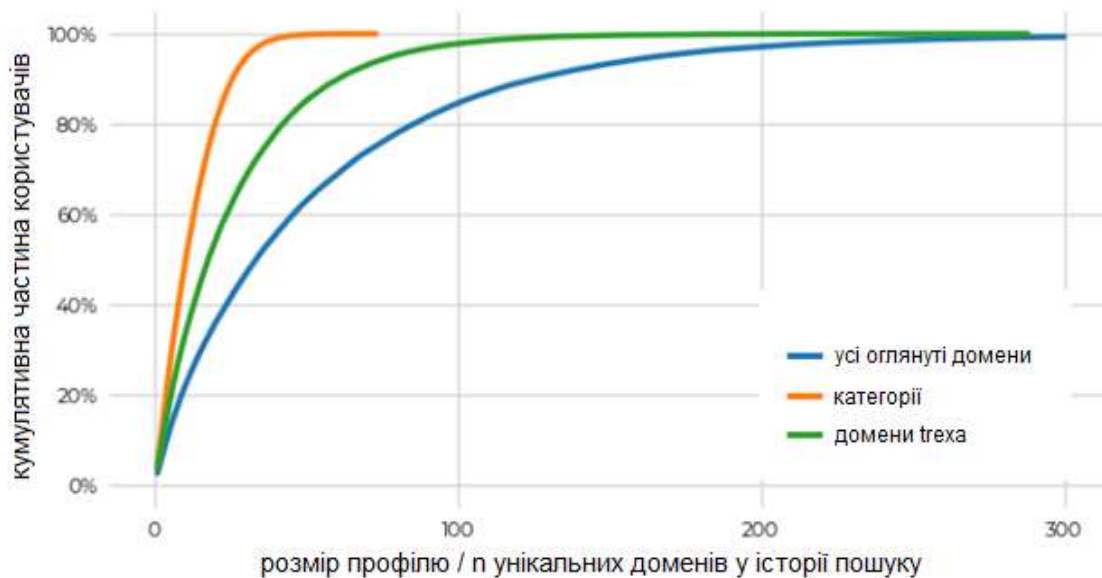


Рисунок 1.11 – Сукупна частка користувачів у розмірі профілю

Коли кількість користувачів з певним профілем досягає 1, називаємо цей профіль називається унікальним, оскільки він спостерігається лише для одного користувача в рамках цієї колекції даних. Повторно виконавши аналіз з кількома підвекторами історії домену, відбираючи вибірки до записів рангу k для різних значень k . У всіх трьох наборах даних $k = 50$ достатньо для вирівнювання нахилу лінії з таким, який генерується з усієї сукупності. Крім того, довжина відрізка при $y = 1$ для кожного рядка на діаграмі вказує кількість профілів, які вважаються унікальними. Цей висновок узгоджується з результатами, представленими Олейник та іншими незважаючи на відмінності в базових наборах даних.

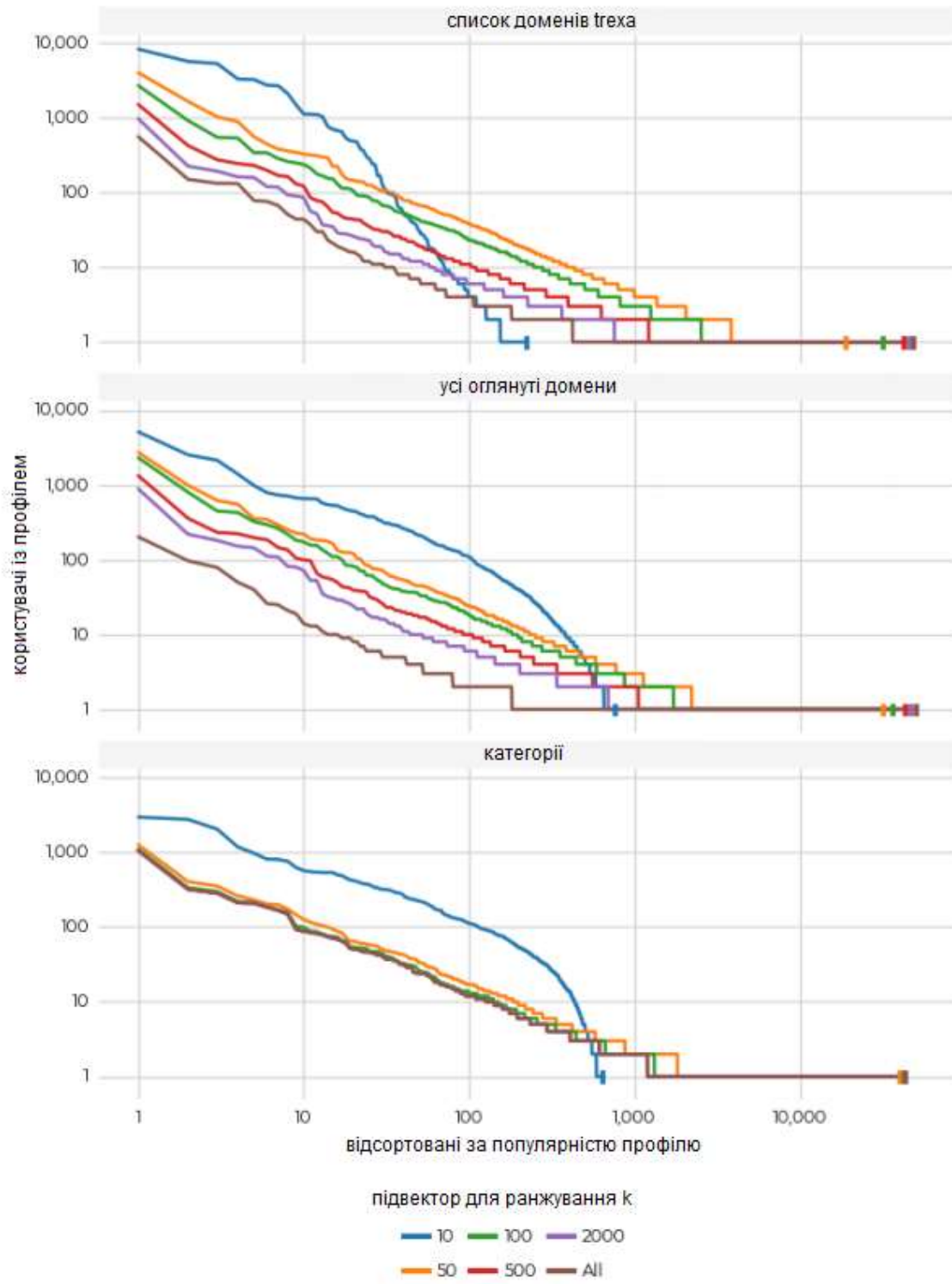


Рисунок 1.12 – Розподіл частоти різних профілів, класифікованих за популярністю, у кількох підвекторах провідних доменів

1.6 Висновки по розділу 1

Отже, розглянувши детально систему Google Analytics було остаточно вирішено про її подальший розгляд та використання. Методи та способи ідентифікації, які використовує Google Analytics будуть розглянуті та порівняні із існуючими вбудованими у браузері системами.

В даному розділі виконано наступні задачі:

- виділено мету дипломної роботи та сформульовано завдання;
- розглянуто систему Google Analytics, результати якої будуть використані у роботі;
- розглянуто системи-аналоги: Amplitude, IO technologies, Segment та Chartbeat, визначено їх переваги та недоліки;
- розглянуто Google Analytics детально для переконання, що можливостей достатньо для виконання необхідної роботи;
- зроблено висновки, щодо систем, для подальшого розвитку власного програмного забезпечення;
- детально розглянуто обрану систему Google Analytics та її можливості;

2 ПЛАНУВАННЯ ЕКСПЕРИМЕНТУ ДОСЛІДУ ПАРАМЕТРІВ ДЛЯ ІДЕНТИФІКАЦІЇ КОРИСТУВАЧІВ У МЕРЕЖІ

Експеримент – це особливий вид діяльності дослідника, який він здійснює з метою пізнання певних властивостей об’єктів або систем та закономірностей які їх пов’язують між собою. При проведенні експерименту (рисунку 2.1) дослідник діє на об’єкт з допомогою спеціальних методів, інструментів або приладів (вектор X) і може планомірно змінювати умови, які необхідні для отримання результатів (вектор Y).

Виявлення властивостей ідентифікації сервісів статистичного збору інформації для забезпечення безпеки інформації.

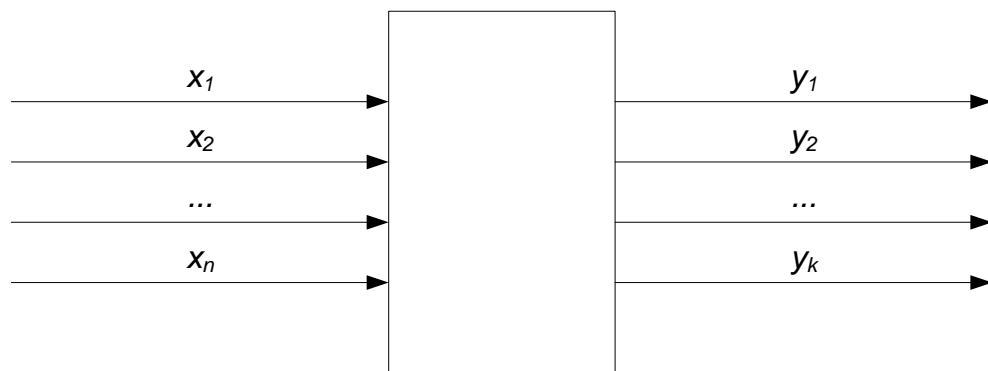


Рисунок 2.1 – Загальна схема проведення експерименту

2.1 Вибір характеристик

2.1.1 Екран та відеопідсистема

`screen.availWidth screen.availHeight screen.colorDepth;`

наприклад: 1366x1024x32

`screen.width screen.height screen.colorDepth;`

наприклад: 1024x1366x32

`screen.orientation.angle + "°"`

Зняття відбитків з холсту працює за рахунок використання елемента холсту HTML5 [5].

Коли користувач відвідує сторінку, скрипт зняття відбитку спочатку рисує текст із обраним шрифтом та розміром та додає колір фону (1). Далі скрипт викликає метод ToDataURL Canvas API для отримання даних пікселів холсту у форматі dataURL (2), який в основному представляє собою представлення двійкових даних пікселів у кодуванні Base64. Нарешті скрипт бере хешданих пікселів у текстовому кодуванні (3), які служать у якості відбитку пальця.

Оскільки відбиток пальця в основному заснований на браузері, операційній системі і встановленому графічному обладнанні, він не дозволяє однозначно ідентифікувати користувачів. У дрібномасштабному дослідженні з 294 учасниками з Amazon's Mechanical Turk спостерігалася експериментальна ентропія в 5,7 біт. Автори дослідження припускають, що більше ентропії, ймовірно, можна було б спостерігати в дикій природі і з великою кількістю шаблонів, що використовуються в відбитках пальців. Хоча цього недостатньо для однозначної ідентифікації користувачів, цей відбиток може бути об'єднаний з іншими джерелами ентропії для отримання унікального ідентифікатора. Стверджується, що оскільки цей метод ефективно знімає відбитки пальців з графічного процесора, ентропія «ортогональна» ентропії попередніх методів відбитків пальців браузера, таких як дозвіл екрана і можливості браузера JavaScript. [7]

На рисунку 3.2 зображено приклад зображення, яке приховано створюється у браузері для ідентифікації за допомогою метода Fingerprint.



Рисунок 2.2 – Зображення за яким створюється ідентифікатор fingerprint

2.1.2 Платформа та програмне забезпечення

platform;

наприклад: «iPhone», «Linux x86_64», «MacIntel», «Win32»

window.navigator.vendor;

наприклад: «Apple Computer, Inc.» , «Google Inc.»

navigator.appName;

наприклад: «Netscape»

navigator.appCodeName;

наприклад: «Mozilla»

navigator.appVersion;

наприклад: «5.0 (Linux; Android 10; ELE-L29) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.86 Mobile Safari/537.36»

2.1.3 Розгляд Cookies та їх можливостей

Файл cookie – це текстовий файл, який використовується для:

- зберігати інформацію про уподобання користувачів, місцезнаходження та інші деталі.
- захистити дані користувачів від несанкціонованого доступу.
- підтримувати функціональність певних веб-сайтів
- обслуговувати персоналізовану рекламу для користувачів та робити рекламу більш ефективною шляхом ремаркетингу.
- збирати дані Google Analytics та інші дані відстеження.

Файли cookie зберігаються на жорсткому диску користувачів і їх можна ввімкнути або вимкнути за допомогою налаштувань веб-браузера.

Бібліотека JavaScript analytics.js встановлює такі власні файли cookie:

- `_ga` cookie, який використовується для ідентифікації унікальних користувачів, і термін його дії закінчується через 2 роки;

– `_gat` cookie, який використовується для зменшення частоти запитів, і термін його дії закінчується через 10 хвилин.

Файли cookie встановлюються в домені верхнього рівня, щоб користувачі могли відстежувати їх у всіх субдоменах без додаткової конфігурації.

Для створення файлів `cookie analytics.js` потрібно створити об'єкт відстеження за допомогою команди «create».

Синтаксис: `ga («create», «UA-XXXX-Y», «auto»);`

Файли `cookie analytics.js` встановлюються, як тільки користувач завантажує вашу веб-сторінку у свій браузер (за умови, що веб-сторінка містить дійсний універсальний код відстеження аналітики).

Щоб переглянути файли `cookie analytics.js`, необхідно виконати наведені нижче дії.

1 Відвідайте веб-сторінку, яка містить дійсний універсальний код відстеження аналітики, у своєму браузері `chrome`.

2 Натисніть кнопку меню `Chrome` на панелі інструментів браузера, а потім виберіть `Налаштування`:

3 Прокрутіть сторінку вниз, а потім унизу натисніть посилання «Показати розширені налаштування».

4 Натисніть кнопку «Налаштування вмісту».

5 Натисніть кнопку «Усі файли cookie та дані веб-сайту...».

6 Знайдіть ім'я домену, веб-сторінку якого ви відвідали в кроці 1, а потім натисніть на текст «cookies»:

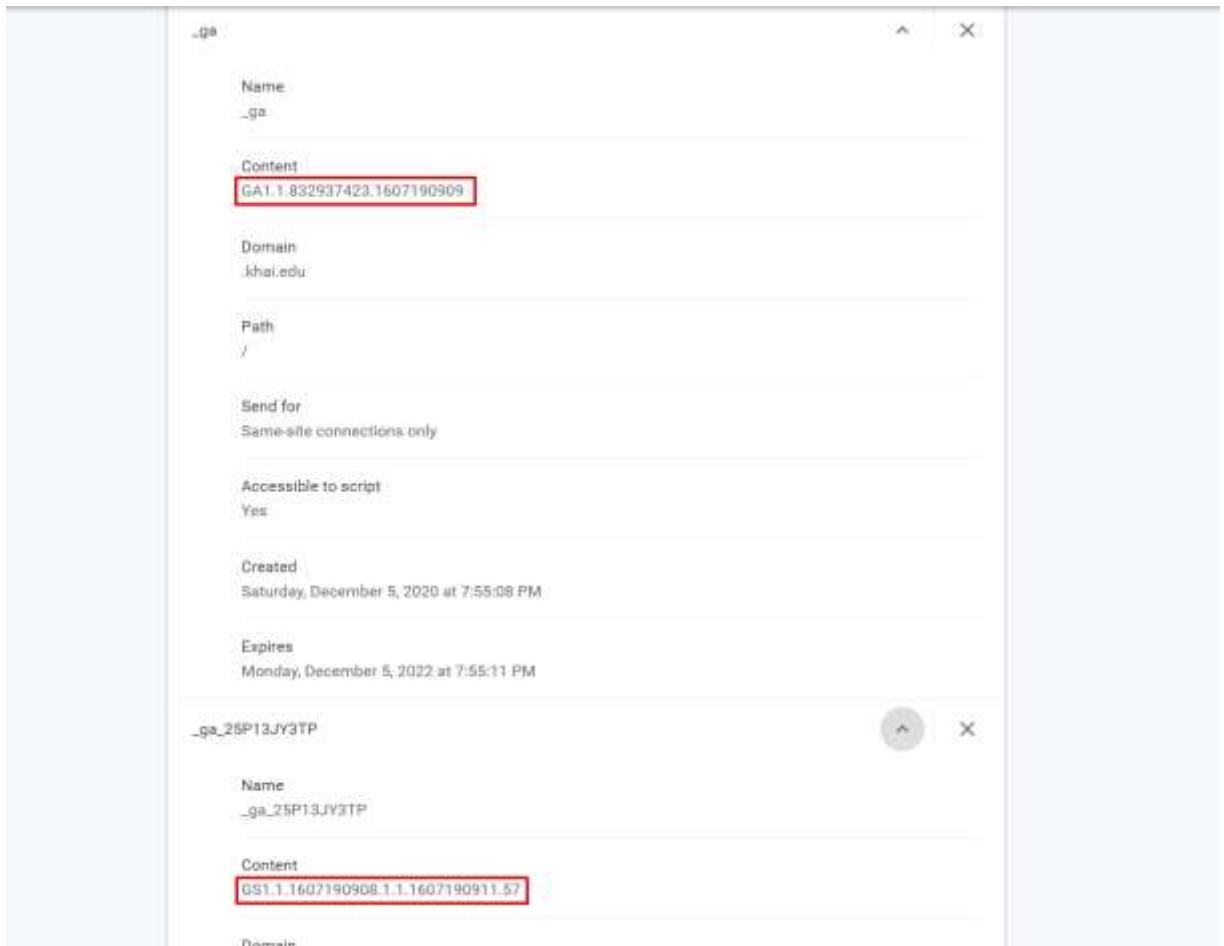


Рисунок 2.3 – Відображення файлів Cookie

2.2 Застосування hash-функцій

Хешування в цілому – це корисний спосіб зменшити величезну кількість даних до короткого (ish) числа, яке можна використовувати для ідентифікації цього зображення. Хешування – це функція, яка застосовується до довільним даними і виробляє дані фіксованого розміру (в основному дуже малого розміру). Існує багато різних типів хеш, але якщо ми говоримо про хешуванні зображень, то воно використовується або для того, щоб:

Знаходити дублікати дуже швидко. Майже будь-яка функція hash буде працювати. Замість того щоб шукати весь образ цілком, ви будете шукати hash цього образу.

2.3 Уточнення умов проведення експерименту

Обрано наступні фактори для проведення експерименту:

- X1 браузер;
- X2 режим роботи браузера;
- X3 версія браузера;
- X4 пристрій;
- X5 операційна система.

На рисунку схематично зображена схема для експерименту.

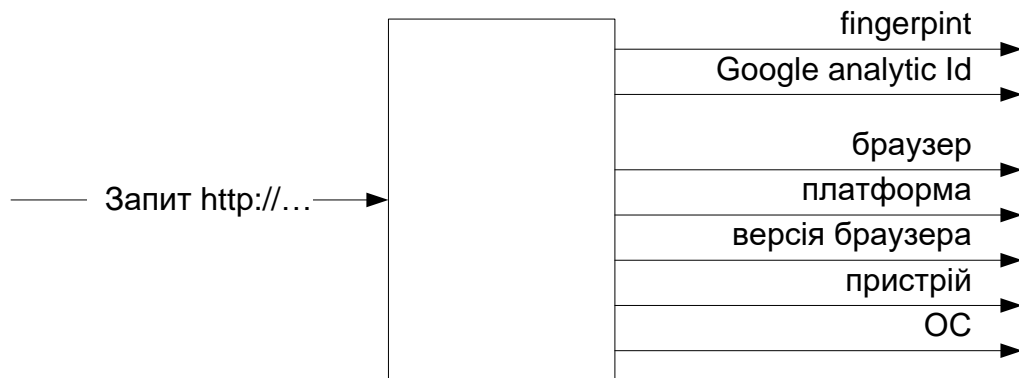


Рисунок 2.4 – Схема експерименту

2.4 Детальний розгляд обраної системи: Google Analytics

Google Analytics – це чудова послуга з багатьох причин, не останньою з яких є її нульова ціна. Ще однією великою перевагою для власників сайтів є простота налаштування. Всього за кілька простих кроків власники сайтів можуть відстежувати більшість (але не всіх) відвідувачів свого сайту.

Після реєстрації в обліковому записі та додавання невеликого фрагмента коду на кожну сторінку власники сайтів готові відстежувати велику кількість інформації. Але не кожного відвідувача можна відстежити, і не всі типи взаємодії можна записати за допомогою базової інсталяції Google Analytics. І причини обох цих обмежень зумовлені внутрішньою роботою самої GA.

Дані з публікації Тіма Марко «How Does Google Analytics Actually Work?» [55] допоможе демістифікувати те, що насправді відбувається в Google Analytics. Розглянемо як GA збирає інформацію від відвідувачів сайту. З кращим розумінням цього процесу можна буде розпізнавати ситуації, коли Google Analytics може не надавати сайту точну інформацію.

2.4.1 Як Google Analytics збирає інформацію про відвідувачів?

Принцип роботи Google Analytics полягає в зборі інформації про відвідувачів сайту. Від джерел трафіку веб-сайту до кількості звернень на місяць, Google Analytics пропонує чудову інформацію про поведінку користувачів і збирає цінні дані користувачів у режимі реального часу.

GA використовує мову програмування під назвою Javascript для передачі інформації на сервери Google про відвідувачів сайту. Майже двадцять років Javascript - це (майже) універсальна мова, яка дозволяє кодерам та розробникам робити сайти інтерактивними.

Ви можете знати, що коли ви бачите веб-сайт, текст, зображення та інша інформація, що міститься на сторінці, зберігається у форматі HTML, який ваш браузер декодує та відображає. Простий спосіб подумати над цим полягає в тому, що HTML схожий на формат файлу .doc, що використовується для документів Microsoft Word. Як і документ Word, файл HTML може сказати комп'ютеру, який текст відображати, де включати зображення та таблиці та яке форматування використовувати.

Але на відміну від документа Word, HTML-сторінки доступні лише для читання. Це означає, що без допомоги такої мови, як Javascript, сторінка HTML не дозволить вам взаємодіяти з веб-сайтом. Наприклад, ви не можете змінити текст та комп'ютер або зв'язатися із сервером.

Javascript обходить це обмеження, оскільки це стандартизована мова, тобто більшість браузерів можуть приймати інструкції, написані на Javascript, і

забезпечувати додаткову інтерактивність. Завдяки Javascript можливо все - від меню вильоту до спливаючих вікон.

Але що важливо для Google Analytics, Javascript дозволяє вашому комп'ютеру надсилати та отримувати повідомлення на інші комп'ютери, підключені до Інтернету. Коли ви відвідуєте веб-сайт, який впровадив Google Analytics, він попросить ваш комп'ютер тимчасово завантажити деякі інструкції Javascript із Google Analytics.

Цей код, у свою чергу, попросить ваш браузер надіслати деякі важливі фрагменти даних на сервери Google. Дані можуть містити конкретну веб-сторінку, яку ви переглядаєте, технічну інформацію про ваш комп'ютер, спосіб вашого прибуття на сайт або спосіб навігації по сайту.

Дослідження показали, що близько 50% з мільйонів веб-сайтів, що входять до Інтернету, використовують GA. Це означає, що більшу частину часу, коли ви перебуваєте в Інтернеті, ваша інформація надсилається на сервери Google. Це може здатися вам трохи тривожним, оскільки існують очевидні проблеми щодо конфіденційності, коли стільки інформації збирається під час перегляду веб-сторінок.

На щастя, Google сприйняв це занепокоєння настільки серйозно, що воно вбудовано в сам дизайн GA. Замість того, щоб збирати інформацію про конкретного користувача та прив'язувати її до свого імені, GA доручає надсилати її інформацію анонімно. Якщо людина заходить на сайт і перевіряє три сторінки, GA не збиратиме інформацію у формі «Тім Марко тричі відвідував ваш сайт». Натомість він просто повідомляє сайту, що користувач, використовуючи певне ключове слово, браузер та з певного регіону, відвідував їхній сайт тричі.

2.4.2 Отримання даних від Google Analytics

2.4.2.1 Використання GA як інструменту для організації та збору інформації в експерименті

Відстеження активних користувачів на кроки 1, 7, 14 і 28 днів отримання даних користувачів щодо сайту.

Тобто якщо користувач повертався за ці проміжки часу – він буде з тим самим ідентифікатором.

2.4.2.2 Провідник користувачів

2.4.2.2.1 Мова та місцезположення

На рисунку 2.9 зображена інформація відносно мови ідентифікованих користувачів.

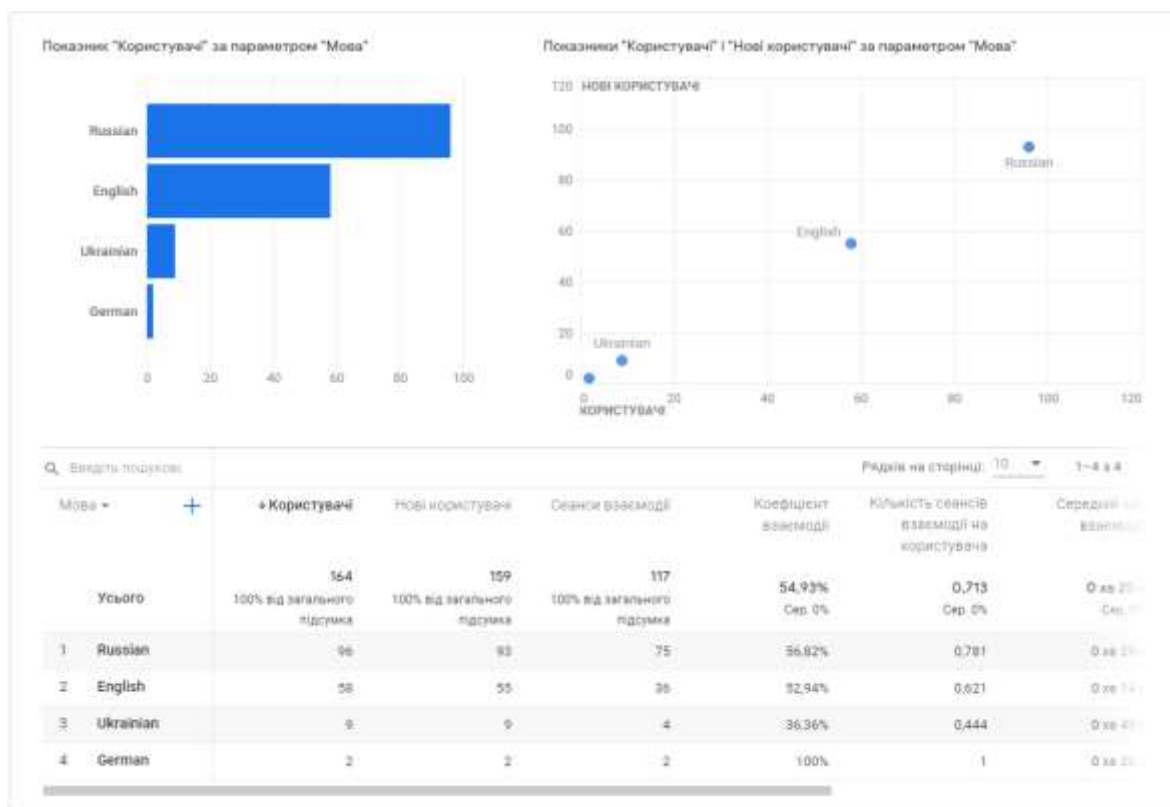


Рисунок 2.5 – Дані ідентифікованих користувачів за параметром «Мова»

2.4.2.2.2 Додаткові маркери

Рисунок 2.10 дозволяє дізнатися деякими інтересами користувачів взагалі, але не надає інформації про конкретних користувачів.

Інтереси		Користувачі	Нові користувачі	Сесії взаємодії	Коефіцієнт взаємодії
Усього		38 100% від загального підсумка	257 100% від загального підсумка	213 100% від загального підсумка	62,83% Сер. 0%
1	Shoppers/Value Shoppers	22	21	16	57,14%
2	Media & Entertainment/Gamers	19	18	13	54,17%
3	Media & Entertainment/Movie Lovers	19	19	14	66,67%
4	Media & Entertainment/Music Lovers	18	17	14	56%
5	Technology/Technophiles	18	17	14	60,87%
6	Food & Dining/Fast Food Cravers	16	15	13	61,9%
7	Home & Garden/Do-It-Yourselfers	14	14	12	66,67%
8	Lifestyles & Hobbies/Shutterbugs	14	13	13	66,42%
9	Food & Dining/Cooking Enthusiasts/30 Minute Chefs	13	13	7	50%
10	Lifestyles & Hobbies/Nightlife Enthusiasts	12	12	11	64,71%
11	Media & Entertainment/Light TV Viewers	12	11	10	66,67%
12	Food & Dining/Foodies	11	11	9	75%
13	Lifestyles & Hobbies/Business Professionals	11	10	10	83,33%
14	Lifestyles & Hobbies/Pet Lovers	11	11	6	42,86%
15	Media & Entertainment/TV Lovers	11	11	7	53,85%

Рисунок 2.6 – Дані користувачів щодо їх інтересів

2.4.3 Налаштування GA для проведення експерименту

Google Analytics налаштовується невеликим фрагментом коду. На рисунку 2.7 зображено власне налаштування у GA.

URL-АДРЕСА ПОТОКУ
https://ejudge.khai.edu/ejudge/id-test

НАЗВА ПОТОКУ
id-test

ІДЕНТИФІКАТОР ПОКАЗНИКА
G-25P13JY3TP

ІДЕНТИФІКАТОР ПОТОКУ
2189713255

СТАН
Трафік надходить протягом останніх 48 годин. [Докладніше](#)

Розширена статистика

Автоматичне відстеження взаємодій і вмісту на ваших сайтах на додачу до стандартного показника переглядів сторінки. Дані елементів на сторінці, як-от посилання і оновлені відео, можуть збиратися з релевантними подіями. Ви несете відповідальність за те, щоб у Google не надсилюлась ідентифікаційна інформація. [Докладніше](#)

Вимірювання: **Перегляди сторінки** **Прокручування** **Кліки зовнішніх посилань** + ще 3

Вказівки з додавання тегів
Налаштуйте свій веб-сайт одним із наведених нижче способів, щоб надіслати дані в цей потік.

[Додати новий тег на сторінці](#) [Використовувати наявний тег сторінки](#)

Глобальний тег сайту (gtag.js)

Скопіюйте та вставте цей код на початку тегу **<head>** для кожної веб-сторінки, яку потрібно відстежувати.

```
<!-- Global site tag (gtag.js) - Google Analytics -->
<script async src="https://www.googletagmanager.com/gtag/js?id=G-25P13JY3TP"></script>
<script>
  window.dataLayer = window.dataLayer || [];
  function gtag(){dataLayer.push(arguments);}
  gtag('js', new Date());

  gtag('config', 'G-25P13JY3TP');
</script>
```

Менеджер тегів Google
Додавайте теги й керуйте ними через веб-інтерфейс, щоб надіслати дані в Google Analytics, а також інші інструменти Google і сторонніх розробників.

Рисунок 2.7 – Налаштування GA для власного експерименту

2.4.3.1 Представлення даних

Представлення даних – це точка доступу користувача до звітів і певний набір даних про ресурс. Користувачам можна надавати доступ до представлення даних, щоб вони могли переглядати звіти з відповідною статистикою. Ресурс може містити одне або кілька представлень даних.

Наприклад, для одного ресурсу можна створити:

- одне представлення всіх даних для www.example.com;
- одне представлення лише даних трафіку AdWords до www.example.com;
- одне представлення лише даних трафіку до субдомену, наприклад, www.sales.example.com.

Коли додається в обліковий запис ресурс, Analytics створює для нього перше представлення даних. Воно не має фільтрів, а тому містить усі дані ресурсу. Важливо залишити його без змін. Можна створювати додаткові представлення й застосовувати до них фільтри, щоб у звіті показувалися лише потрібні вам дані.

У звіті за вибраним представленням даних буде подано статистику за період із дати створення представлення. Наприклад, якщо представлення даних було створено 1 червня, можна переглянути лише статистику, отриману після цієї дати.

Якщо видалити представлення даних, відповідний набір даних буде втрачено. Не видаляйте представлення даних, якщо в майбутньому вам може знадобитися звіт за цим набором даних.

2.4.3.2 Користувачі й дозволи

Є можливість додавати користувачів в обліковий запис Analytics. Це можна робити на рівні облікового запису, ресурсу або представлення даних, а також установлювати обмеження для користувачів на кожному з цих рівнів. Щоб додати користувача, потрібно вказати електронну адресу його облікового запису Google і призначити йому відповідні дозволи. Відповідно до отриманих дозволів користувач зможе керувати іншими користувачами, виконувати завдання адміністративного характеру (наприклад, створювати додаткові представлення даних і фільтри), а також переглядати дані у звітах [29].

2.4.4 Імпорт даних

Імпортування даних дає змогу завантажувати дані із зовнішніх джерел і об'єднувати їх із даними, зібраними через Analytics. Відтак за допомогою Analytics усі ці дані можна впорядкувати й аналізувати в найзручніший для бізнесу спосіб.

Імпортування дає змогу об'єднати дані автономних бізнес-систем із даними, які Analytics збирає онлайн. Ця операція допоможе створити й проаналізувати об'єднаний набір даних, а також приймати на його основі рішення, які краще відповідають унікальним потребам вашого бізнесу.[14]

2.4.5 Звіти від Google Analytics

Спеціальний звіт – це звіт, який користувач створив сам, обираючи параметри та показники, а потім вирішує, як вони мають відображатися.

Для створення звіту необхідно:

- 1 Увійти у Google Analytics.
- 2 Відкрити потрібне представлення даних.
- 3 Відкрити Звіти.
- 4 Натиснути Налаштування > Спеціальні звіти > +Новий спеціальний звіт.
- 5 Ввести назву.
- 6 Натиснути +додати вкладку звіту *(необов'язково)*. Кожен звіт містить принаймні 1 вкладку, однак можна додати більше (Рисунок 1.10).
- 7 Вибрати тип звіту: Огляд, Таблиця з однорівневою адресацією, Накладання даних на карту або Послідовність.
- 8 *Натиснути, щоб розгорнути й дізнатися більше про типи звітів.*
- 9 Визначити параметри й показники. *Параметри для кожного типу звітів різняться, оскільки кожен тип відображає дані по-іншому.*
- 10 Щоб створити звіт тільки за певними параметрами *(необов'язково)*, натиснути +додати фільтр.
- 11 *Натиснути, щоб розгорнути й дізнатися більше про фільтри.*
- 12 Указати, де має відображатися цей звіт *(необов'язково)*. Вибрати потрібні варіанти в спадному меню або ввімкніть параметр Усі представлення даних, які пов'язані з цим обліковим записом, щоб включити всі представлення, до яких у вас є доступ.

13 Натиснути Зберегти.[15]

Create Custom Report

General Information

Title

Report Content

Report Tab ×

+ add report tab

Name

Type Explorer Flat Table Map Overlay Funnel BETA

Рисунок 2.8 – Створення звіту

2.4.6 Структура звіту про відвідування

В усіх звітах про відвідування (окрім звіту "Візуалізація послідовності") використовуються однакові базові компоненти.

2.4.6.1 Вузли

Вузли – це точки, через які проходить трафік. Вузлом можуть бути різні елементи, як-от окремі сторінки чи екрани, каталоги, групи вмісту, події чи параметри. Кожен вузол містить число, яке вказує обсяг трафіку (кількість користувачів), що через нього пройшов.

Вузли параметрів позначаються білим кольором. Вузол параметра визначає базову точку входу в потік, а тому ви можете порівняти, як впливають на потік користувачів різні канали маркетингу, соціальні засоби комунікації, параметри технологій та інші точки входу.

Вузол параметра

Вузли сторінок, груп вмісту та цілей і екрани додатків позначаються зеленим кольором.

Вузол сторінки чи групи вмісту (рисунок 2.6)



Рисунок 2.9 – Вузол цілі

Вузли подій позначаються синім кольором (рисунок 2.7).



Рисунок 2.10 – Вузол події

2.4.6.2 Зв'язки, переривання та повернення

Зв'язки відображаються як сірі смуги між вузлами та позначають шлях, пройдений сегментом трафіку від одного вузла до іншого. Товщина зв'язку вказує на відносний обсяг трафіку в цьому сегменті: що товща смуга, то більше трафіку проходить через це зв'язок.

Іноді ці шляхи несподівано повертаються до попередніх вузлів або обходять якісь вузли. Повернення відображаються як зв'язок зі стрілкою, спрямованою на попередній вузол.

Переривання (інша назва – виходи) вказують на місце виходу користувача з потоку.

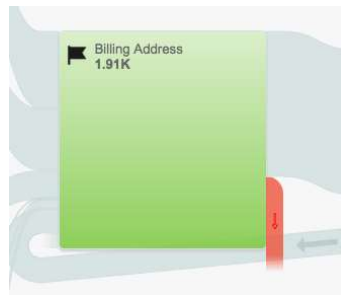


Рисунок 2.11 – Зв'язки, входи та виходи та повернення

2.4.7 Використання звітів про відвідування

Кожен звіт про відвідування містить дані для певного сегмента Analytics, але в усіх звітах ви знайдете дані про:

- відносний обсяг трафіку до вашого сайту за вибраним параметром (наприклад, джерелами трафіку, кампанією, веб-переглядачем);
- відносну кількість користувачів на кожному кроці (вузлі) шляху;
- відносний обсяг трафіку й показник відмов між кроками (виходи);
- місця на шляху, де користувачі повертаються до попередніх вузлів;
- окремі показники для зв'язків, вузлів і виходів із вузлів (якщо навести на них курсор).

2.4.8 Обмеження звітів про відвідування

2.4.8.1 Обмеження сеансів

Звіти про відвідування можуть містити дані про щонайбільше 100 000 сеансів. Якщо набір даних міститиме більше сеансів, до звітів буде застосовано вибірку. Відкоригуйте діапазон дат, щоб зменшити кількість сеансів у звіті.

2.4.8.2 Обмеження сегментів

Звіти про відвідування підтримують лише сегменти на основі сеансів. Сегменти на основі користувачів для цього типу звітів вимкнено. До звіту про відвідування можна застосувати лише один сегмент [32].

2.4.9 Налаштування доступу до даних

В обліковому записі Analytics є кілька параметрів доступу до даних. Вони дають змогу налаштувати спосіб надсилання в Google даних, зібраних за допомогою різних методів Analytics (наприклад, коду відстеження JavaScript, пакетів SDK для мобільних пристроїв і протоколу Measurement Protocol). Так ви можете керувати доступом до даних відповідно до своїх потреб. Ці параметри дають змогу налаштувати доступ лише до тих даних, які зібрано з веб-сайтів, мобільних додатків та інших цифрових пристроїв за допомогою Analytics. Вони не застосовуються до даних про ваш обліковий запис Analytics, його використання, кількість ресурсів і налаштовані додаткові функції. Незалежно від налаштувань доступу до даних, дані Analytics можуть використовуватися лише, якщо це потрібно для підтримки та захисту служби Analytics.

Усі представники Google (включно з постачальниками), які мають доступ до даних облікових записів, повинні прийняти загальні положення та умови політики внутрішнього доступу. Доступ до даних потребує відповідної автентифікації, здійснюється через протокол SSL і реєструється у відповідних журналах системи безпеки. Представники Google можуть отримувати доступ до даних користувачів лише на затверджених компанією Google комп'ютерах [30].

2.4.10 Про ідентифікаційні дані в контрактах і правилах Google

У багатьох контрактах, правилах і умовах використання продуктів Google для аналітики й показу реклами є положення щодо ідентифікаційної інформації або ідентифікаційних даних. Ці положення забороняють надсилати в Google

інформацію, яку Google може використовувати або класифікувати як ідентифікаційну.

Це інформація, за допомогою якої можна ідентифікувати особу, зв'язатися з нею або точно визначити її місцеположення. Така інформація включає:

- електронні адреси;
- поштові адреси;
- номери телефонів;
- точні місцеположення, як-от GPS-координати (однак дивіться примітку нижче);
- повні імена та прізвища або імена користувачів.

Натомість Google не вважає ідентифікаційними даними:

- псевдонімні ідентифікатори файлів cookie;
- псевдонімні рекламні ідентифікатори;
- IP-адреси;
- інші псевдонімні ідентифікатори кінцевих користувачів.

Наприклад, якщо IP-адреса надсилається із запитом оголошення (а так відбувається майже завжди у зв'язку з використанням інтернет-протоколів), це не вважатиметься порушенням заборони на надання ідентифікаційних даних у Google [31].

2.5 Статистичний аналіз

Метою аналізу даних є знання про об'єкт дослідження – виявлення корисної інформації, знайдення висновків, врешті, розумне (зважене) прийняття рішень. Аналіз даних може мати багато аспектів та підходів, реалізовуватися за Метою аналізу даних є знання про об'єкт дослідження – виявлення корисної інформації – параметри пристрою користувача, що отримуються через запит, знайдення висновків та розумне (зважене) прийняття рішень – вирішення чи можливе використання отриманих даних надалі. Аналіз

даних може мати багато аспектів та підходів, реалізовуватися за допомогою різних інструментів – в тому числі математичних, статистичних, за допомогою різноманітних способів візуалізації, але головне завдання його – стиснення інформації.

Працюючи з відкритими даними, часто можна мати справу масивами даних, що містять сотні, тисячі, а то й сотні тисяч спостережень. Неозброєним оком людина навряд чи зможе зробити з цих даних якісь коректні висновки. Тому потрібно застосовувати різні засоби для узагальнення, стиснення інформації, представленої в «сирих» або мікроданих.

Статистика – наука збору, представлення, аналізу і розумної інтерпретації даних.

Статистика являє собою строгий науковий метод, що дозволяє зрозуміти дані, дійти до їх суті. Наприклад, у дослідженні вивчаються дані відвідувань сайту 50 користувачів-учасників. Це уже достатньо велика кількість спостережень, і просто глянувши на дані неможливо отримати швидке інформативне уявлення. Однак статистика може дати миттєву загальну картину даних – на основі доступної для сприйняття візуалізації або числового узагальнення – незалежно від кількості спостережень чи одиниць даних. Окрім узагальнення даних, важливим завданням статистики є формулювання логічних висновків та передбачення залежностей між змінними.

2.6 Висновки з розділу 2

Розглянувши більш детально можливості Google Analytics було вирішено додати також використання ідентифікації за допомогою Murmur хешування, а результати обробляти за допомогою статистичного аналізу.

Було розглянуто використання звітів Google Analytics для подальшої обробки результатів.

Також було виконано налаштування збору даних у Google Analytics для використання лише релевантної інформації.

3 ПРОВЕДЕННЯ ЕКСПЕРИМЕНТУ Й АНАЛІЗ РЕЗУЛЬТАТІВ

3.1 Проведення експерименту

На рисунку 3.1 наведено схему проведеного експерименту ідентифікації користувачів у мережі. Для проведення експерименту було опитано приблизно 50 людей. Наступним етапом був збір необхідних даних для подальшого аналізу.

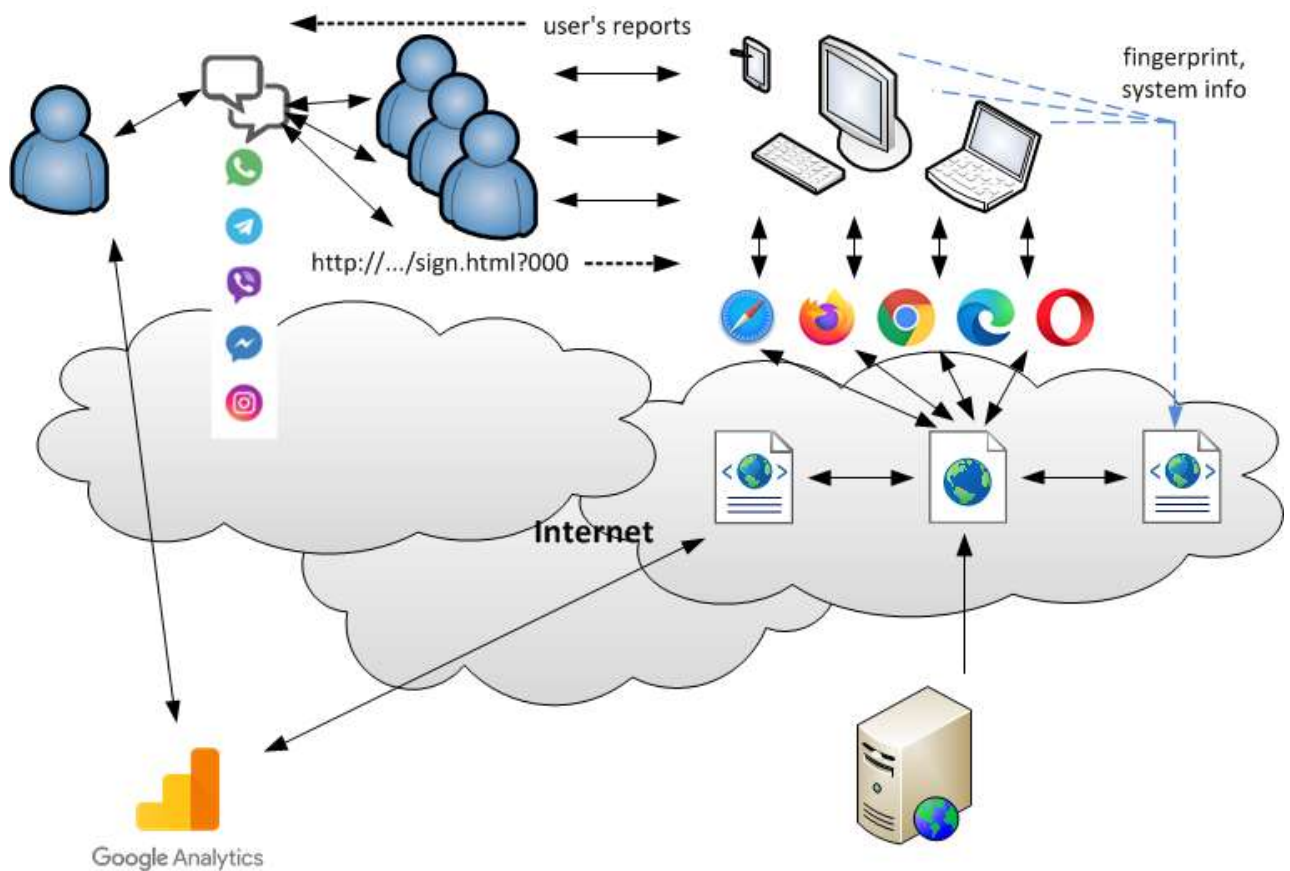


Рисунок 3.1 – Схема експерименту

У таблиці 3.1 наведено статистику даних зібраних від опитаних користувачів.

Таблиця 3.1 – Загальні характеристики отриманих результатів

№	Параметр	Опис параметру	Кількість відповідей	
			всього	унікальних
1	#	Номер учасника в рамках експерименту	160	50
2	Platform	Назва апаратної платформи	160	7
3	Vendor	Розробник інструментарію веб-браузера	143	3
4	Screen	Роздільна здатність екрану	160	44
5	Window	Роздільна здатність робочої області	160	56
6	gaID	Ідентифікатор користувача, сформований GA	67	54
7	fpID	Ідентифікатор користувача, сформований за методом Fingerprint	160	76
8	Version	Версія встановленого у користувача програмного забезпечення	160	69
9	GA Events	Звернень до тестової сторінки (за даними GA)	173	
10	GA Id	Унікальних ідентифікаторів GA	157	
11	GA #	Зафіксована кількість користувачів (за даними GA)	53	

Кількість відповідей з GA Id 31,2%

Кількість співпадінь GA Id у відповідях та звітах 22,3%

Один пристрій 26,0%

Два пристрої 70,0%

Більше двох пристроїв 4,0%

3.1.1 Формування коду ідентифікації на основі методу Fingerprint

Нижче наведено сформований код ідентифікації на основі методу Fingerprint

```
var strng = "";

function fingerprint() {
    var canvas = document.createElement('canvas');
    canvas.width = 350;
    canvas.height = 50;
    var ctx = canvas.getContext('2d');
    var txt = '53AC2950-66C2-4D22-996E-8CBD0FFF5C77\ud83d\ude03';
    ctx.textBaseline = "top";
    ctx.font = "16px 'Arial'";
```

```

ctx.textBaseline = "alphabetic";
ctx.rotate(.05);
ctx.fillStyle = "#f60";
ctx.fillRect(125, 1, 162, 20);
ctx.fillStyle = "#069";
ctx.fillText(txt, 2, 15);
ctx.fillStyle = "rgba(102, 200, 0, 0.7)";
ctx.fillText(txt, 4, 17);
ctx.shadowBlur = 10;
ctx.shadowColor = "blue";
ctx.fillRect(-20, 10, 364, 5);
strng = canvas.toDataURL();

var hash = MurmurHash3(strng).result();

return hash;
}

```

У Додатку А наведено фрагмент використаного коду murmurhash3.

3.1.2 Організація взаємодії з Google Analytics

Взаємодія із Google Analytics організована через фрагмент коду наведений нижче.

```

<!DOCTYPE html>
<html>
<head>
  <!-- Global site tag (gtag.js) - Google Analytics -->
  <script async src="https://www.googletagmanager.com/gtag/js?id=G-25P13JY3TP"></script>

  <script>
    window.dataLayer = window.dataLayer || [];
    function gtag() { dataLayer.push(arguments); }
    gtag('js', new Date());
    gtag('config', 'G-25P13JY3TP');

    function getcId() {
      var match = document.cookie.match("(?:^|;)\s*_ga=([^\;]*)");
      var raw = (match) ? decodeURIComponent(match[1]) : null;
      if (raw) {
        match = raw.match(/(\d+\.\d+)$/);
      }
      var gacid = (match) ? match[1] : null;
      if (gacid) {
        return gacid;
      }
    }
  </script>

  <meta charset="utf-8" />
  <title></title>

  <script type="text/javascript" async src="fp.js"></script>
</head>

```

```

<body>
  <div id="platform"></div>
  <div id="vendor"></div>
  <div id="scr01"></div>
  <div id="scr02"></div>
  <div id="scr03"></div>
  <div id="cid"></div>
  <div id="fpid"></div>
  <div id="appname"></div>
  <div id="appcode"></div>
  <div id="appver"></div>
  <div id="ref"></div>
  <div id="fpimg" style="border-left: 6px solid red; background-color:
lightgrey; width:350px;"></div>
  <script>
    document.getElementById("cid").textContent = getcId();
    document.getElementById("ref").textContent = window.location.href;
    document.getElementById("platform").textContent =
window.navigator.platform;
    document.getElementById("vendor").textContent = window.navigator.vendor;
    document.getElementById("scr02").textContent = screen.availWidth + "x" +
screen.availHeight + "x" + screen.colorDepth;
    document.getElementById("scr01").textContent = screen.width + "x" +
screen.height + "x" + screen.colorDepth;
    document.getElementById("appname").textContent = navigator.appName;
    document.getElementById("appcode").textContent = navigator.appCodeName;
    document.getElementById("appver").textContent = navigator.appVersion;
    document.getElementById("scr03").textContent = screen.orientation.angle
+ ""
  </script>

</body>
</html>

```

На рисунку 3.2 наведено приклад відгуку, який учасники отримують в результаті виконання коду для збору експериментальних даних.



Рисунок 3.2 – Приклад результатів експерименту

3.2 Отримані дані від Google Analytics

Google Analytics дозволяє спостерігати дані у реальному часі за останні 30 хвилин (рисунок 3.4). Можна побачити кількість користувачів, тип пристрою з якого було виконано перехід за посиланням, а також країну звідки людина переходила за посиланням.

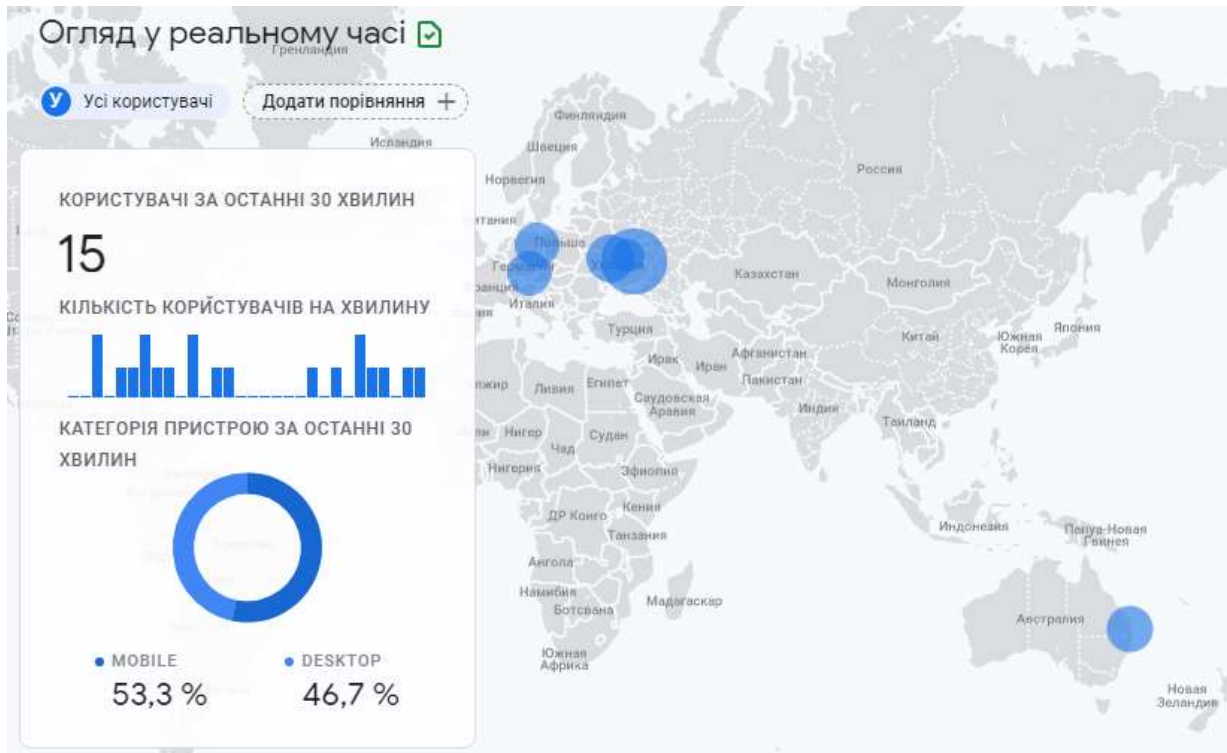


Рисунок 3.3 – Дані користувачів у реальному часі

Відстеження активних користувачів на кроки 1, 7, 14 і 28 (30) днів отримання даних користувачів щодо сайту. На рисунку 3.4 наведено графік активності користувачів за певні періоди часу

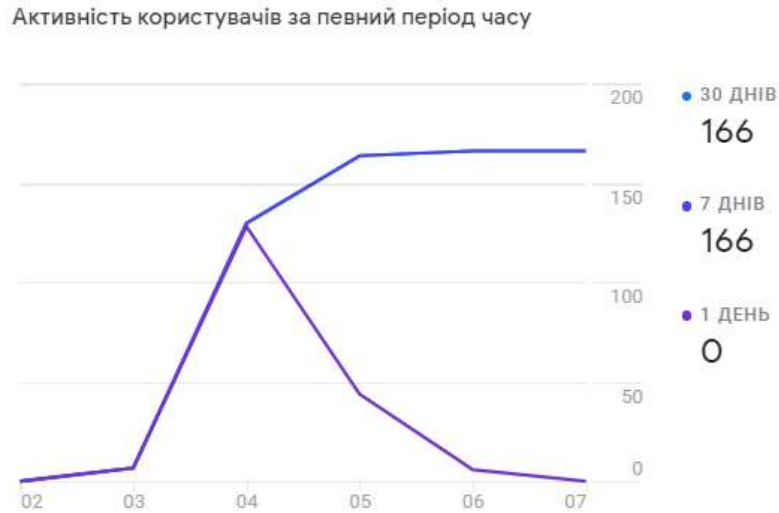


Рисунок 3.4 – Графік активності під час проведення експерименту

Тобто якщо користувач повертався за ці проміжки часу – він буде з тим самим ідентифікатором.

На рисунку 3.5 зображено географічне положення користувачів експерименту



Рисунок 3.5 – Географічне положення учасників експерименту

3.3 Murmur хешування

Murmur – це сімейство хороших функцій хешування загального призначення, придатних для некриптографічного використання та алгоритм MurmurHash забезпечує наступні переваги:

- простий (за кількістю згенерованих інструкцій);
- хороший розподіл (проходження тестів χ -квадрат практично для всіх наборів ключів;
- гарна лавиноподібна поведінка;
- хороша стійкість до зіткнень;
- відмінна продуктивність на обладнанні Intel / AMD, хороший компроміс між якістю hash і споживанням CPU.

Можна використовувати його для hash UUIDs (зменшити зображення fingerprint до 32 біт (хешірованное значення). Незалежно від якості функції хешування, будуть виникати колізії але використання інженерної функції hash, такий як Murmur, дозволить максимізувати якість розподілу і звести до мінімуму кількість зіткнень, але іншої гарантії не дає. Обраний спосіб хешування наведено у Додатку А.

3.4 Пояснення отриманих результатів

Google Analytics збрає усі дані до звіту та надає його у вигляді таблиць (Рисунок 3.6).

Є два важливі висновки щодо того, як GA збирає інформацію, що має значення для будь-якого власника сайту.

Перший полягає в тому, що, оскільки він написаний на Javascript, деякі користувачі завжди будуть невидимі для GA. Як згадувалось раніше, Javascript є майже універсальною мовою, але факт залишається фактом: деякі користувачі

або не мають можливостей Javascript, або активно вимикають Javascript у своїх браузерах.

ID	Дата	Назва сторінки	Кількість переглядів	Сторінка
Всього				
			2350	212
			100% від загальної кількості переглядів	100% від загальної кількості сторінок
1	20201204	id-test	312	1
2	20201204	id-test	311	2
3	20201204	id-test	175	4
4	20201204	id-test	165	5
5	20201204	id-test	140	6
6	20201204	id-test	96	1
7	20201204	id-test	41	1
8	20201204	id-test	34	2
9	20201204	id-test	34	1
10	20201204	id-test	31	1
11	20201204	id-test	29	2
12	20201204	id-test	25	1
13	20201204	id-test	25	2
14	20201204	id-test	25	2
15	20201204	id-test	22	1
16	20201204	id-test	19	1

Рисунок 3.6 – Формування звіту у системі GA із отриманням ідентифікаторів учасників

Сучасні комп'ютери, планшети та смартфони майже всі мають функцію Javascript, встановлену за замовчуванням. Все ще існує чимала кількість старих систем, які все ще використовуються, які не використовують Javascript. Як результат, вони ніколи не відобразяться у звітах Analytics.

Подібним чином деякі досвідченіші користувачі можуть змінити свої налаштування, щоб запобігти коли-небудь впливу Javascript на їх браузер (як правило, через проблеми конфіденційності або продуктивності). Отже, якщо з якоїсь причини вам потрібно мати повністю точну інформацію про те, скільки разів переходили на ваш сайт, GA не збирається різати гірчицю.

З іншого боку, GA може пропустити великий сегмент вашого трафіку, якщо цільова аудиторія або надзвичайно підкована (за допомогою спеціальних комп'ютерів), або не підкована (за допомогою застарілих технологій). Якщо ви стурбовані тим, що це стосується вашого сайту, вам, мабуть, слід розглянути рішення на основі серверного журналу для доповнення відстеження вашого сайту.

Іншим важливим висновком є те, що завдяки використанню анонімного відстеження GA не дозволяє глибоко відстежувати взаємодію на рівні користувача. Для деяких веб-сайтів, особливо тих, які вимагають від користувачів реєстрації та входу, важливо розуміти, як окремі користувачі переходять зі сторінки на сторінку. Так само важливо знати або чим один сеанс відрізнявся від іншого.

Хоча можна отримати частину цієї інформації із налаштувань GA за замовчуванням, відстежувати користувачів, які входять із декількох пристроїв, може бути набагато складніше. У такому випадку, можливо, варто розглянути таке рішення, як Piwik.

Загалом, два обмеження Google Analytics, про які згадувались раніше, насправді не є головними недоліками для більшості сайтів. Існує маса причин, чому програмне забезпечення використовується половиною провідних веб-сайтів у світі.

3.5 Аналіз отриманих даних

Важливим параметром порівняння різних систем ідентифікації є коефіцієнт надійності – ймовірність помилок 1-го і 2-го роду.

Помилка першого роду (FRR – False Rejection Rate) – це ймовірність помилкового відмови в доступі клієнту, який має право доступу. Дана помилка може з'являтися при підвищенні порога чутливості (в системах, де даний параметр регульований) або при розбіжності ідентифікатора. Людина нервує, його доступ утруднений, а в деяких випадках швидкий доступ є важливим параметром.

Помилка другого роду (FAR – False Acceptance Rate) - це ймовірність помилкового доступу, коли система помилково пізнає чужого як свого. Для бізнес-систем критичного призначення даний параметр є найбільш

«критичним», оскільки область застосування систем ідентифікації – об'єкти з підвищеними вимогами щодо безпеки.

Також системи ідентифікації іноді характеризуються коефіцієнтом рівній ймовірності помилок 1-го і 2-го роду (EER – Equal Error Rates), що представляє точку збігу ймовірностей FRR і (іноді званого Crossover Equal Error Rates). Якісна і надійна і система повинна мати низький рівень EER.

У деяких системах існує можливість регулювання порогу чутливості, що дозволяє гнучко їх налаштовувати відповідно до вимог з безпеки. Не слід, однак, забувати, що, наприклад, збільшення чутливості системи (і, як наслідок, зниження ймовірності помилкового доступу – FAR) одночасно супроводжується збільшенням часу ідентифікації і підвищенням ймовірності помилкового відмови – FRR. У цьому сенсі найбезпечніша система та, яка завжди відмовляє в доступі.

У таблиці 3.2-3.4 наведено узагальнення унікальності ідентифікаторів, комбінацій ідентифікаторів та порівняння унікальності ідентифікаторів на різних платформах.

Таблиця 3.2 – Узагальнення унікальності ідентифікаторів

№	Характеристика	Platform	Vendor	Screen	Window	gaID	fpID	Version
1	Унікальність ідентифікатора	0,0%	0,0%	61,4%	71,4%	100,0%	71,1%	82,6%
2	Дублювання ідентифікатора	100,0%	100,0%	38,6%	28,6%	0,0%	28,9%	17,4%
3	Кількість користувачів з одним ідентифікатором	26,0%	76,0%	24,0%	24,0%	32,0%	14,0%	20,0%
4	Кількість користувачів з кількома ідентифікаторами	74,0%	20,0%	76,0%	76,0%	28,0%	86,0%	80,0%
5	Кількість користувачів без ідентифікатора	0,0%	4,0%	0,0%	0,0%	40,0%	0,0%	0,0%

Таблиця 3.3 – Порівняння унікальності ідентифікаторів на різних платформах

№	Platform	Унікальність ідентифікатора	Дублювання ідентифікатора
1	iPhone	8,9%	1,6%
2	Linux aarch64	2,4%	0,0%
3	Linux armv7l	4,0%	0,0%
4	Linux armv8l	30,6%	0,0%
5	Linux x86_64	3,2%	0,0%
6	MacIntel	7,3%	0,0%
7	Win32	41,1%	0,8%

Таблиця 3.4 – Оцінка ефективності

№	Параметр	показник помилкової відмови, FRR	показник помилкового пропуску, FAR
1	Fingerprint	41.3%	25.0%
2	Google analytics	1.9%	0.0%
3	Screen (Display)	30.0%	33.8%
4	Window (work space)	30.6%	26.9%
5	Version	38.8%	26.9%
6	Platform	25.0%	51.9%
7	Fingerprint+Version	43.8%	8.8%
8	Fingerprint+Window	45.6%	5.6%
9	Fingerprint+Window+Version	48.1%	1.9%

3.6 Практичне використання результатів експерименту

Для практичного використання був використаний шаблон контролю працездатності (Sanity Check Pattern) – це дуже легкий шаблон, що забезпечує мінімальне покриття помилок (рисунок 3.7). Мета шаблону – переконатися, що система функціонує в рамках допустимого більш-менш робить щось розумне, навіть якщо не зовсім правильне. Це корисно в ситуаціях, коли спрацьовування не є критичним при правильному виконанні (наприклад, при додатковому удосконаленні), але може завдати шкоди, якщо воно виконано неправильно. Це варіант шаблону «монітор-виконавець» і, як і цей шаблон, передбачає наявність відказостійкого стану.

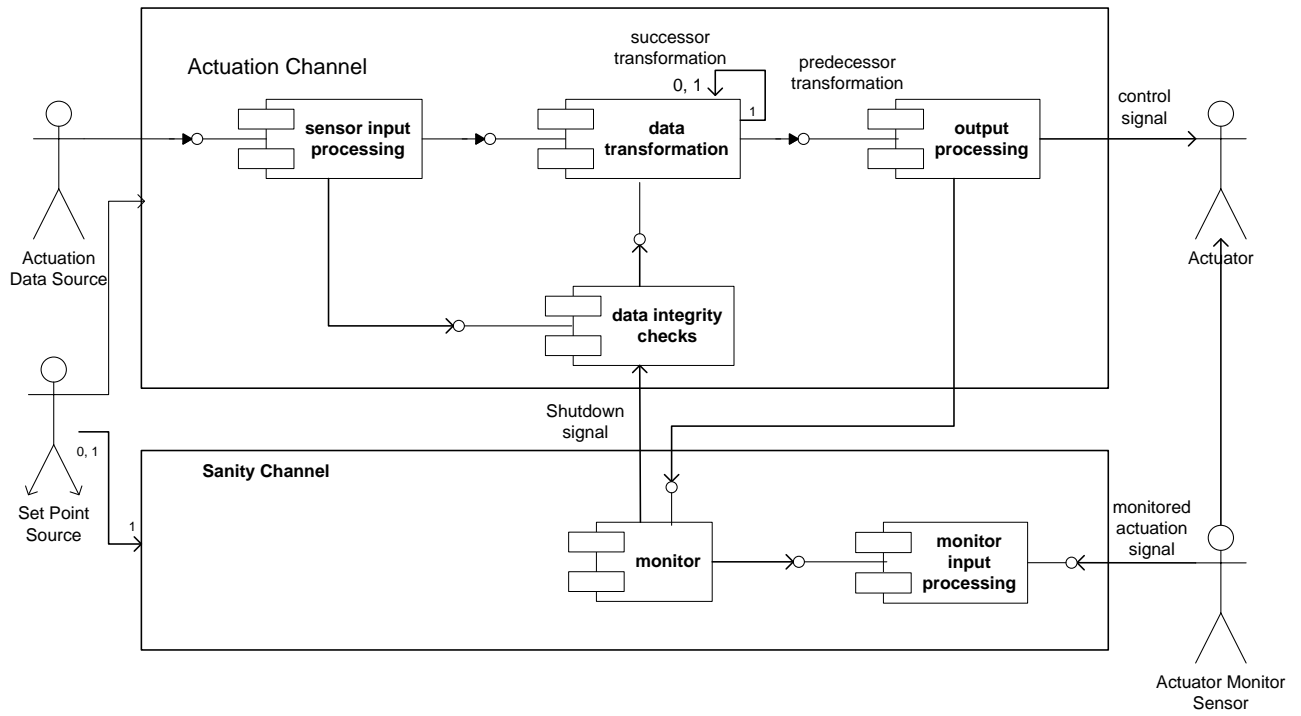


Рисунок 3.7 – Шаблон Sanity Check

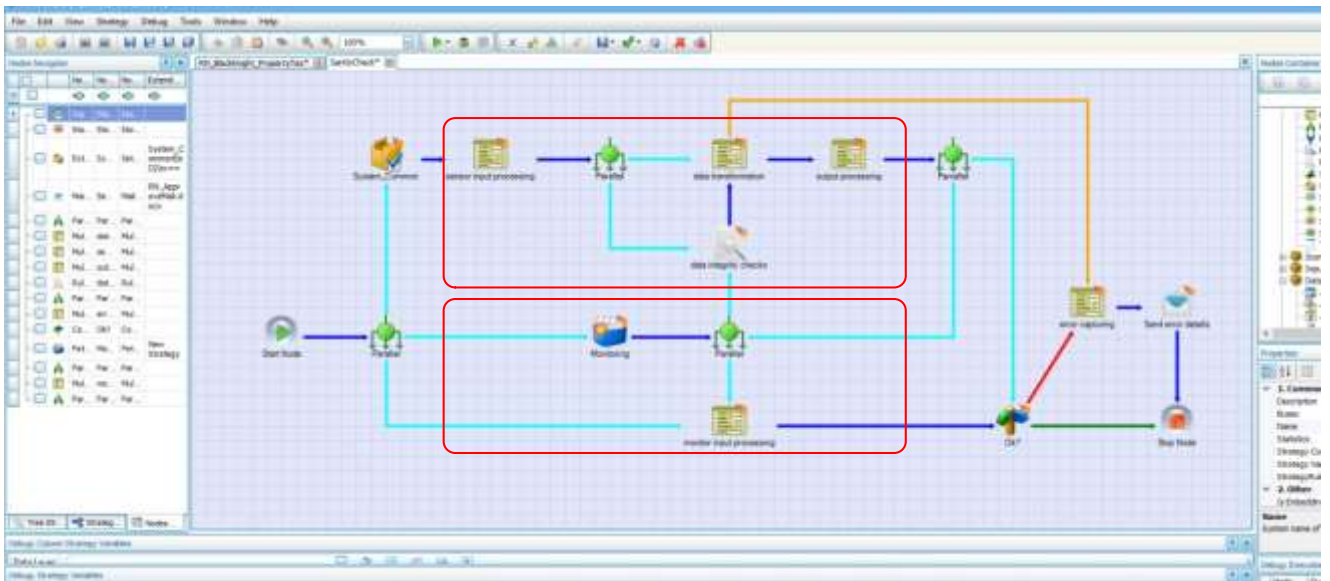


Рисунок 3.8 – Шаблон Sanity Check

3.7 Висновки по розділу 3

Даний розділ присвячено проведенню експерименту та обробці даних методом статистичного аналізу. Було наведено приклади отриманих даних та проаналізовано їх.

Результати дещо відрізнялись від очікуваних, але це не вплинуло на остаточний результат аналізу.

ВИСНОВОК

У даній дипломній роботі на тему «Дослідження методів і способів ідентифікації користувачів на основі було виконано наступне:

- оглянуто різні системи обліку користувачів на основі мережевої активності, такі як Google Analytics, Amplitude, IO technologies, Segment та Chartbeat та виконано їх порівняння, а також було розглянуто сучасний спосіб збору інформації через розширення браузера Mozilla;

- було сплановано експеримент та більш детально розглянуто можливості роботи із Google Analytics та з її звітами, а також обрано статистичний метод аналізу даних для експерименту та виконано налаштування звітів у Google Analytics для майбутньої обробки даних;

- проведено експеримент із приблизно 50 людьми та зібрано дані від кожного з них, в цей же час Google Analytics збирала дані на кожен перехід за посиланням, яке було надано учасникам експерименту, результати експерименту було проаналізовано та виконано порівняння за різними факторами.

Отже, використання Google Analytics не надасть повного підтвердження ідентифікації користувача, так само як і метод fingerprint'у. Підвищити ефективність ідентифікації можливо за допомогою поєднання різних дозволених способів та методів ідентифікації

ПЕРЕЛІК ПОСИЛАНЬ

- 1 Google Analytics [Електронний ресурс] – Режим доступу до ресурсу: <https://startpack.ru/application/google-analytics>.
- 2 Конверсія [Електронний ресурс] – Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/Конверсія>.
- 3 Amplitude [Електронний ресурс] – Режим доступу до ресурсу: <https://startpack.ru/application/amplitude>.
- 4 IO technologies [Електронний ресурс] – Режим доступу до ресурсу: <https://startpack.ru/application/onthe-io>.
- 5 Segment [Електронний ресурс] – Режим доступу до ресурсу: <https://startpack.ru/application/segment>.
- 6 What is Google Analytics and how does it work [Електронний ресурс] – Режим доступу до ресурсу: <https://www.antevenio.com/usa/what-is-google-analytics-and-how-does-it-work/>.
- 7 SimilarWeb PRO [Електронний ресурс] – Режим доступу до ресурсу: <https://startpack.ru/application/similarweb-traffic-analysis>.
- 8 Content Insights [Електронний ресурс] – Режим доступу до ресурсу: <https://startpack.ru/application/contentinsights>.
- 9 Giraff [Електронний ресурс] – Режим доступу до ресурсу: <https://startpack.ru/application/giraff>.
- 10 Методи ідентифікації, системи ідентифікації. Біометричні технології [Електронний ресурс] – Режим доступу до ресурсу: <https://studfile.net/preview/5129599/>.
- 11 Типи ідентифікації користувачів [Електронний ресурс] – Режим доступу до ресурсу: <http://www.infobezpeka.com/publications/?id=92>.
- 12 Інтегрована інформаційно-аналітична система: проект та концепція [Електронний ресурс] – Режим доступу до ресурсу: https://pidruchniki.com/1031062453915/informatika/integrovana_informatsiyno-analitichna_sistema_proekt_kontseptsiya.

13 Конверсії [Електронний ресурс] – Режим доступу до ресурсу: <https://support.google.com/analytics/answer/1012040?hl=uk>.

14 Про імпортування даних [Електронний ресурс] – Режим доступу до ресурсу:

https://support.google.com/analytics/answer/3191589?hl=uk&ref_topic=6064627.

15 Створення спеціальних звітів і керування ними [Електронний ресурс] – Режим доступу до ресурсу:

https://support.google.com/analytics/answer/1151300?hl=uk&ref_topic=1012046.

16 Analytics for Publishers, from Startup to Enterprise [Електронний ресурс] – Режим доступу до ресурсу: <https://iotechnologies.com>.

17 Amplitude [Електронний ресурс] – Режим доступу до ресурсу: <https://amplitude.com>.

18 Segment [Електронний ресурс] – Режим доступу до ресурсу: <https://segment.com>.

19 Chartbeat: Content Analytics and Insights for Digital Publishing [Електронний ресурс] – Режим доступу до ресурсу: <https://chartbeat.com>.

20 Сравнение сервисов Segment и Google Analytics [Електронний ресурс] – Режим доступу до ресурсу: <https://startpack.ru/compare/google-analytics/segment>.

21 Сравнение сервисов Amplitude и Google Analytics [Електронний ресурс] – Режим доступу до ресурсу: <https://startpack.ru/compare/google-analytics/amplitude>.

22 Сравнение сервисов Google Analytics и IO technologies [Електронний ресурс] – Режим доступу до ресурсу: <https://startpack.ru/compare/google-analytics/onthe-io>.

23 Сравнение сервисов Google Analytics и Chartbeat [Електронний ресурс] – Режим доступу до ресурсу: <https://startpack.ru/compare/chartbeat/google-analytics>.

24 Chande S. Google Analytics -Case study [Електронний ресурс] / Suraj Chande – Режим доступу до ресурсу:

https://www.researchgate.net/publication/271447580_Google_Analytics_-_Case_study.

25 Erturk E. A case study in open source software security and privacy: Android adware [Електронний ресурс] / Emre Erturk – Режим доступу до ресурсу:

https://www.researchgate.net/publication/261348136_A_case_study_in_open_source_software_security_and_privacy_Android_adware.

26 A practical evaluation of Web analytics [Електронний ресурс] – Режим доступу до ресурсу:
до ресурсу:

https://www.researchgate.net/publication/235316271_A_practical_evaluation_of_Web_analytics.

27 Plaza B. Monitoring web traffic source effectiveness with Google Analytics [Електронний ресурс] / Beatriz Plaza – Режим доступу до ресурсу:
до ресурсу:

https://www.researchgate.net/publication/235306326_Monitoring_web_traffic_source_effectiveness_with_Google_Analytics.

28 Nakatani K. A Web Analytics Tool Selection Method: an Analytical Hierarchy Process Approach [Електронний ресурс] / K. Nakatani, T. Chuang – Режим доступу до ресурсу:
до ресурсу:

https://www.researchgate.net/publication/220146697_A_Web_Analytics_Tool_Selection_Method_an_Analytical_Hierarchy_Process_Approach.

29 Ієрархія організацій, облікових записів, користувачів, ресурсів і представлень даних [Електронний ресурс] – Режим доступу до ресурсу:
до ресурсу:

https://support.google.com/analytics/answer/1009618?hl=uk&ref_topic=3544906.

30 Налаштування доступу до даних [Електронний ресурс] – Режим доступу до ресурсу:
до ресурсу:

https://support.google.com/analytics/answer/1011397?hl=uk&ref_topic=2919631.

31 Про ідентифікаційні дані в контрактах і правилах Google [Електронний ресурс] – Режим доступу до ресурсу:
до ресурсу:

https://support.google.com/analytics/answer/7686480?hl=uk&ref_topic=1008008.

32 Звіти з візуалізацією відвідувань [Електронний ресурс] – Режим доступу до ресурсу: https://support.google.com/analytics/answer/2519986?hl=uk&ref_topic=6207573.

33 TRANCO: A Research-Oriented Top Sites Ranking Hardened Against Manipulation [Електронний ресурс] / [V. Le Pochat, T. V. Goethem, S. Tajalizadehkhoob та ін.]. – 2018. – Режим доступу до ресурсу: <https://arxiv.org/pdf/1806.01156.pdf>.

34 Libert T. An automated approach to auditing disclosure of third-party data collection in website privacy policies. [Електронний ресурс] / Timothy Libert // Proceedings of the 2018 World Wide Web Conference. – 2018. – Режим доступу до ресурсу: <https://arxiv.org/pdf/1805.01187.pdf>.

35 Mozilla. Mozilla Privacy Policy [Електронний ресурс]. – 2018. – Режим доступу до ресурсу: <https://www.mozilla.org/en-US/privacy/>.

36 Mozilla. Introducing firefox pioneer [Електронний ресурс]. – 2019. – Режим доступу до ресурсу: <https://addons.mozilla.org/en-US/firefox/pioneer>.

37 Mozilla. Jestr pioneer shield study. [Електронний ресурс]. – 2019. – Режим доступу до ресурсу: <https://github.com/mozilla/jestr-pioneer-shield-study>

38 Englehard S. Online tracking: A 1-million-site measurement and analysis [Електронний ресурс] / S. Englehard, A. Narayanan // Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. – 2016. – Режим доступу до ресурсу: https://www.cs.princeton.edu/~arvindn/publications/OpenWPM_1_million_site_tracking_measurement.pdf.

39 The web never forgets: Persistent tracking mechanisms in the wild. [Електронний ресурс] / [G. Acar, C. Eubank, S. Englehard та ін.] // Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security. – 2014. – Режим доступу до ресурсу: https://securehomes.esat.kuleuven.be/~gacar/persistent/the_web_never_forgets.pdf.

40 Bahia K. Internet Society Global Internet Report: Consolidation in the Internet Economy [Електронний ресурс] / K. Bahia, S. Suardi // The Internet Society. – 2019. – Режим доступу до ресурсу: https://future.internetsociety.org/2019/?gclid=Cj0KCQiAkuP9BRckARIsAKGLE8Xx_Gu4ZMuD29eQP9FhvzqH5EbG2juHd3Z_B_lbJVzgitS3t6OeOCBwaApNtEALw_wcB.

41 Banse C. Tracking users on the internet with behavioral patterns: Evaluation of its practical feasibility [Электронный ресурс] / C. Banse, D. Herrmann, H. Federrath // IFIP Advances in Information and Communication Technology. – 2012. – Режим доступа до ресурсу: https://link.springer.com/chapter/10.1007/978-3-642-30436-1_20.

42 Gómez-Boix A. Hiding in the crowd: An analysis of the effectiveness of browser fingerprinting at large scale [Электронный ресурс] / A. Gómez-Boix, P. Laperdrix, B. Baudry // Proceedings of the 2018 World Wide Web Conference, WWW '18. – 2018. – Режим доступа до ресурсу: <https://hal.inria.fr/hal-01718234v2>.

43 Pretorius S. Attributing users based on web browser history [Электронный ресурс] / S. Pretorius, A. R. Ikuesan, H. S. Venter // In 2017 IEEE Conference on Application, Information and Network Security (AINS). – 2017. – Режим доступа до ресурсу: <https://ieeexplore.ieee.org/document/8270427>.

44 The representativeness of automated web crawls as a surrogate for human browsing [Электронный ресурс] / [D. Zeber, S. Bird, C. Oliveira та ін.] // Proceedings of the 29th International Conference on World Wide Web, WWW '20. – 2020. – Режим доступа до ресурсу: https://research.mozilla.org/files/2020/02/Jestr_vs_crawl_WWW20202.pdf.

45 Analyzing characteristic host access patterns for re-identification of web user sessions [Электронный ресурс] / D.Herrmann, C. Gerber, C. Banse, H. Federrath // Information Security Technology for Applications, NordSec 2010. – 2010. – Режим доступа до ресурсу: https://link.springer.com/chapter/10.1007/978-3-642-27937-9_10.

46 Roesner F. Detecting and defending against third-party tracking on the web [Электронный ресурс] / F. Roesner, T. Kohno, D. Wetherall // Presented as part of the 9th USENIX Symposium on Networked Systems Design and Implementation (NSDI 12). – 2012. – Режим доступа до ресурсу: <https://www.usenix.org/conference/nsdi12/technical-sessions/presentation/roesner>.

47 Iqbal U. The ad wars: Retrospective measurement and analysis of antiadblock filter lists. [Электронный ресурс] / U. Iqbal, Z. Shafiq, Z. Qian // Proceedings of the 2017 Internet Measurement Conference, IMC '17. – 2017. – Режим доступа до ресурсу: <https://conferences.sigcomm.org/imc/2017/papers/imc17-final113.pdf>.

48 Zeber D. Top site list comparison analysis [Электронный ресурс] / David Zeber – Режим доступа до ресурсу: <https://github.com/mozilla/research-repo-webconf-crawl-representativeness>

49 Webshrinker. Webshrinker apis [Электронный ресурс] // 2020 – Режим доступа до ресурсу: <https://www.webshrinker.com/apis/>.

50 If you are not paying for it, you are the product: How much do advertisers pay to reach you? [Электронный ресурс] / P. Papadopoulos, N. Kourtellis, P. R. Rodriguez, N. Laoutaris // Proceedings of the 2017 Internet Measurement Conference, IMC '17. – 2017. – Режим доступа до ресурсу: <https://arxiv.org/abs/1701.07058>.

51 The Design and Implementation of the Tor Browser [DRAFT] [Электронный ресурс] / M. Perry, E. Clark, S. Murdoch, G. Korpen. – 2019. – Режим доступа до ресурсу: <https://2019.www.torproject.org/projects/torbrowser/design/>.

52 Schelter S. On the ubiquity of web tracking: Insights from a billion-page web crawl / S. Schelter, J. Kunegis. // The Journal of Web Science. – 2018. – №4. – С. 53–66.

53 Firefox Machine Learning Team. Trexa. [Электронный ресурс] – Режим доступа до ресурсу: <https://github.com/mozilla/trexa>.

54 Browser history re:visited [Электронный ресурс] / [M. Smith, C. Disselkoen, S. Narayan та ін.] // In 12th {USENIX} Workshop on Offensive Technologies ({WOOT} 18). – 2018. – Режим доступа до ресурсу: <https://www.usenix.org/conference/woot18/presentation/smith>.

55 Marco T. How Does Google Analytics Actually Work? [Электронный ресурс] / Tim Marco // Digital Third Coast – Режим доступа до ресурсу: <https://www.digitalthirdcoast.com/blog/how-does-google-analytics-actually-work>.

ДОДАТОК А

Функція хешування зображення

```

/**
 * @preserve
 * JS Implementation of incremental MurmurHash3 (r150) (as of May 10, 2013)
 *
 * @author <a href="mailto:jensyt@gmail.com">Jens Taylor</a>
 * @see http://github.com/homebrewing/brauhaus-diff
 * @author <a href="mailto:gary.court@gmail.com">Gary Court</a>
 * @see http://github.com/garycourt/murmurhash-js
 * @author <a href="mailto:aappleby@gmail.com">Austin Appleby</a>
 * @see http://sites.google.com/site/murmurhash/
 */
(function () {
    var cache;

    // Call this function without `new` to use the cached object (good for
    // single-threaded environments), or with `new` to create a new object.
    //
    // @param {string} key A UTF-16 or ASCII string
    // @param {number} seed An optional positive integer
    // @return {object} A MurmurHash3 object for incremental hashing
    function MurmurHash3(key, seed) {
        var m = this instanceof MurmurHash3 ? this : cache;
        m.reset(seed)
        if (typeof key === 'string' && key.length > 0) {
            m.hash(key);
        }

        if (m !== this) {
            return m;
        }
    };

    // Incrementally add a string to this hash
    //
    // @param {string} key A UTF-16 or ASCII string
    // @return {object} this
    MurmurHash3.prototype.hash = function (key) {
        var h1, k1, i, top, len;

        len = key.length;
        this.len += len;

        k1 = this.k1;
        i = 0;
        switch (this.rem) {
            case 0: k1 ^= len > i ? (key.charCodeAt(i++) & 0xffff) : 0;
            case 1: k1 ^= len > i ? (key.charCodeAt(i++) & 0xffff) << 8 : 0;
            case 2: k1 ^= len > i ? (key.charCodeAt(i++) & 0xffff) << 16 : 0;
            case 3:
                k1 ^= len > i ? (key.charCodeAt(i) & 0xff) << 24 : 0;
                k1 ^= len > i ? (key.charCodeAt(i++) & 0xff00) >> 8 : 0;
        }

        this.rem = (len + this.rem) & 3; // & 3 is same as % 4
        len -= this.rem;
        if (len > 0) {
            h1 = this.h1;
            while (1) {

```

```

    k1 = (k1 * 0x2d51 + (k1 & 0xffff) * 0xcc9e0000) & 0xffffffff;
    k1 = (k1 << 15) | (k1 >>> 17);
    k1 = (k1 * 0x3593 + (k1 & 0xffff) * 0x1b870000) & 0xffffffff;

    h1 ^= k1;
    h1 = (h1 << 13) | (h1 >>> 19);
    h1 = (h1 * 5 + 0xe6546b64) & 0xffffffff;

    if (i >= len) {
        break;
    }

    k1 = ((key.charCodeAt(i++) & 0xffff) ^
        ((key.charCodeAt(i++) & 0xffff) << 8) ^
        ((key.charCodeAt(i++) & 0xffff) << 16));
    top = key.charCodeAt(i++);
    k1 ^= ((top & 0xff) << 24) ^
        ((top & 0xff00) >> 8);
}

k1 = 0;
switch (this.rem) {
    case 3: k1 ^= (key.charCodeAt(i + 2) & 0xffff) << 16;
    case 2: k1 ^= (key.charCodeAt(i + 1) & 0xffff) << 8;
    case 1: k1 ^= (key.charCodeAt(i) & 0xffff);
}

    this.h1 = h1;
}

    this.k1 = k1;
    return this;
};

// Get the result of this hash
//
// @return {number} The 32-bit hash
MurmurHash3.prototype.result = function () {
    var k1, h1;

    k1 = this.k1;
    h1 = this.h1;

    if (k1 > 0) {
        k1 = (k1 * 0x2d51 + (k1 & 0xffff) * 0xcc9e0000) & 0xffffffff;
        k1 = (k1 << 15) | (k1 >>> 17);
        k1 = (k1 * 0x3593 + (k1 & 0xffff) * 0x1b870000) & 0xffffffff;
        h1 ^= k1;
    }

    h1 ^= this.len;

    h1 ^= h1 >>> 16;
    h1 = (h1 * 0xca6b + (h1 & 0xffff) * 0x85eb0000) & 0xffffffff;
    h1 ^= h1 >>> 13;
    h1 = (h1 * 0xae35 + (h1 & 0xffff) * 0xc2b20000) & 0xffffffff;
    h1 ^= h1 >>> 16;

    return h1 >>> 0;
};

// Reset the hash object for reuse
//
// @param {number} seed An optional positive integer

```



```
MurmurHash3.prototype.reset = function (seed) {
  this.h1 = typeof seed === 'number' ? seed : 0;
  this.rem = this.k1 = this.len = 0;
  return this;
};

// A cached object to use. This can be safely used if you're in a single-
// threaded environment, otherwise you need to create new hashes to use.
cache = new MurmurHash3();

if (typeof (module) !== 'undefined') {
  module.exports = MurmurHash3;
} else {
  this.MurmurHash3 = MurmurHash3;
}
}());
```

ДОДАТОК Б

Експериментальні дані з Google Analytics

Ідентифікатор копії додатка	Дата	Місто	Країна	Мова	Веб-переглядач	Версія веб-переглядача	Операційна система та її версія	Модель мобільного пристрою	Пристрій	Кількість подій	Сеанси
1013471088.1551551777	20201204	Sumy	Ukraine	Russian	Chrome	87.0.4280.88	Windows 10	Chrome		4	1
1034455622.1607111095	20201204	Kyiv	Ukraine	Ukrainian	Safari	14.0.1	iOS 14.2	iPhone		4	1
1037305486.1607088688	20201204	Dublin	Ireland	English	Firefox	83.0	Linux	Firefox		5	1
1037537196.1580246722	20201204	Kharkiv	Ukraine	English	Chrome	87.0.4280.88	Windows 10	Chrome		8	1
1038791413.1607116826	20201204	Sumy	Ukraine	Russian	Edge	87.0.664.52	Windows 10	Edge		5	1
1079756874.1607115252	20201204	Kharkiv	Ukraine	Russian	Chrome	86.0.4240.198	Windows 10	Chrome		5	1
1114427001.1607090391	20201204	Kharkiv	Ukraine	English	Chrome	86.0.4240.111	Windows 10	Chrome		7	1
1141035791.1555015149	20201204	Kharkiv	Ukraine	Russian	Internet Explorer	11.0	Windows 7	Internet Explorer		239	10
1154050083.1607112580	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.88	Windows 10	Chrome		5	1
1174929747.1607098796	20201204	Leander	United States	English	Chrome	86.0.4240.198	Android 9	SM-G950U	Galaxy S8	4	1
1191788263.1541977410	20201204	Kharkiv	Ukraine	Russian	Chrome	86.0.4240.198	Windows 7	Chrome		166	6
1209199027.1607118618	20201204	Kyiv	Ukraine	Russian	Chrome	86.0.4240.198	Android 10	GM1910	7 Pro	10	1
1238284548.1607101316	20201204	Kyiv	Ukraine	Russian	Chrome	87.0.4280.86	Android 10	Mi 9T		15	1
1238284548.1607101316	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.86	Android 10	Mi 9T		15	1
123958613.1607087136	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.86	Android 9	Redmi Note 5		10	1
1252590762.1607088947	20201204	Kharkiv	Ukraine	Russian	Safari	13.1	iOS 13.4	iPhone		4	1
1280424927.1367851468	20201204	Kharkiv	Ukraine	Russian	Internet Explorer	11.0	Windows 7	Internet Explorer		29	2
130015716.1607089304	20201204	Austin	United States	English	Chrome	86.0.4240.198	Windows 10	Chrome		5	1
1304444746.1607104157	20201204	Kharkiv	Ukraine	Russian	Firefox	83.0	Macintosh Intel 10.14	Safari		6	2
1306058571.1607105217	20201204	Mariupol'	Ukraine	Russian	Chrome	86.0.4240.198	Android 10	EML-L29	P20	5	1
1328669945.1607109586	20201204	Kyiv	Ukraine	Russian	Chrome	57.0.2987.108	Android 6.0	MZ-M5S	M5s	9	1
1363303890.1607111139	20201204	Kyiv	Ukraine	Russian	Chrome	86.0.4240.198	Android 10	Mi 10		4	1
1372173739.1567893189	20201204	Sumy	Ukraine	Russian	Opera	72.0.3815.400	Windows 10	Opera		5	1
1413985239.1597127596	20201204	Kharkiv	Ukraine	English	Chrome	79.0.3945.116	Android 10	SM-M205FN	Galaxy M20	6	1
1415748440.1607117784	20201204	Berlin	Germany	English	Chrome	86.0.4240.198	Android 9	SM-G950F	Galaxy S8	4	1
1424391584.1606839313	20201204	(not set)	Poland	Russian	Chrome	86.0.4240.198	Windows 10	Chrome		5	1
1428803790.1607087189	20201204	(not set)	United States	Russian	Puffin	8.5.0.42207AP,gzip(gfe)	Android 10	Mi 9T		25	1
1428803790.1607087189	20201204	(not set)	United States	Russian	Puffin	8.5.0.42207AP,gzip(gfe)	Linux	Chrome		41	1

Ідентифікатор копії додатка	Дата	Місто	Країна	Мова	Веб-переглядач	Версія веб-переглядача	Операційна система та її версія	Модель мобільного пристрою	Пристрій	Кількість подій	Сеанси
144437943.1607114835	20201204	Austin	United States	English	Chrome	86.0.4240.198	Windows 10	Chrome		5	1
1445935935.1602617732	20201204	Kharkiv	Ukraine	Russian	Chrome	71.0.3578.141	Android 10	Mi 9T		10	1
1456560248.1607115183	20201204	Austin	United States	Russian	Chrome	87.0.4280.88	Windows 10	Chrome		5	1
1457515909.1607101586	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.88	Windows 10	Chrome		29	1
1457832878.1607118874	20201204	Kyiv	Ukraine	Russian	Chrome	86.0.4240.198	Android 10	GM1910	7 Pro	4	1
1510753664.1607100916	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.86	Android 10	Mi 9T		4	1
1513024958.1607098665	20201204	Leander	United States	English	Chrome	86.0.4240.198	Windows 10	Chrome		5	1
1534916846.1548062758	20201204	Kyiv	Ukraine	Russian	Chrome	86.0.4240.198	Android 7.1.1	X00LDA	Zenfone 4 Selfie ZD553KL	6	1
1536696938.1607087096	20201204	Kharkiv	Ukraine	English	Chrome	77.0.3865.90	Windows 10	Chrome		4	1
1547951770.1598866842	20201204	Kyiv	Ukraine	Russian	Chrome	71.0.3578.141	Android 9	Redmi 6A		4	1
157673876.1575643013	20201204	Kharkiv	Ukraine	Russian	Firefox	83.0	Windows 10	Firefox		5	1
1608338372.1607087220	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.86	Android 9	Redmi Note 5		4	1
1640361422.1607090261	20201204	Kharkiv	Ukraine	English	Chrome	87.0.4280.88	Windows 10	Chrome		5	1
1641417820.1607111304	20201204	Kyiv	Ukraine	Russian	Chrome	86.0.4240.198	Android 7.1.1	X00LDA	Zenfone 4 Selfie ZD553KL	4	1
1661108203.1606973814	20201204	Kharkiv	Ukraine	Russian	Chrome	86.0.4240.198	Windows 7	Chrome		308	7
1661108203.1606973814	20201204	Kharkiv	Ukraine	Russian	Safari	11.0	Android 6.0	Nexus 5		34	0
1661108203.1606973814	20201204	Kharkiv	Ukraine	Russian	Safari	13.0.3	iOS 13.2.3	iPhone		24	0
1661761923.1607097852	20201204	Kharkiv	Ukraine	Russian	Chrome	86.0.4240.198	Macintosh Intel 11.0	Safari		5	1
1663323080.1607110958	20201204	Kyiv	Ukraine	Ukrainian	Safari	14.0.1	iOS 14.2	iPhone		6	1
1689492983.1607086997	20201204	Kharkiv	Ukraine	English	Chrome	86.0.4240.198	Android 11	Pixel 4		6	1
1697805427.1607102122	20201204	Kharkiv	Ukraine	Russian	Firefox	83.0	Windows 10	Firefox		5	1
1709207455.1607117799	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.66	Android 8.1.0	ARE-AL00	Honor 8X Max	4	1
171969607.1607102900	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.66	Android 10	Mi 9T		4	1
1727231687.1607088652	20201204	Zaporizhzhia	Ukraine	Ukrainian	Android Webview	67.0.3396.87	Android 6.0.1	(not set)		4	1
1749194371.1607110435	20201204	Kyiv	Ukraine	Russian	Chrome	87.0.4280.66	Android 9	ANE-LX1	P20 Lite	6	1
1753546206.1607117499	20201204	Kyiv	Ukraine	English	Chrome	86.0.4240.198	Android 9	INE-LX1	NOVA 3i	4	1
1781602170.1607086030	20201204	Kharkiv	Ukraine	English	Internet Explorer	11.0	Windows 10	Internet Explorer		4	1

Ідентифікатор копії додатка	Дата	Місто	Країна	Мова	Веб-переглядач	Версія веб-переглядача	Операційна система та її версія	Модель мобільного пристрою	Пристрій	Кількість подій	Сеанси
1801383376.1607086702	20201204	Kyiv	Ukraine	Russian	Chrome	85.0.4183.109	iOS 13.5	iPhone		7	2
1815949238.1601323291	20201204	Kharkiv	Ukraine	English	Opera	72.0.3815.400	Windows 10	Opera		5	1
1861143089.1607116743	20201204	Sumy	Ukraine	Russian	Firefox	83.0	Windows 10	Firefox		4	1
1872751035.1601081563	20201204	(not set)	Netherlands	English	Opera	72.0.3815.400	Windows 7	Opera		34	3
1872751035.1601081563	20201204	Kharkiv	Ukraine	English	Opera	72.0.3815.400	Windows 7	Opera		34	3
1880935632.1589558139	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.77	iOS 14.2	iPhone		4	1
1896188786.1607092572	20201204	Karlsruhe	Germany	German	Firefox	83.0	Windows 10	Firefox		8	1
1929715452.1607114890	20201204	Austin	United States	English	Chrome	86.0.4240.198	Windows 10	Chrome		4	1
1937022894.1567397044	20201204	Kharkiv	Ukraine	English	Chrome	87.0.4280.77	iOS 14.2	iPhone		4	1
1954504288.1607085965	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.66	Windows 10	Chrome		6	1
196712702.1607114197	20201204	Kharkiv	Ukraine	English	Chrome	87.0.4280.88	Windows 7	Chrome		5	1
199345248.1607118524	20201204	Brisbane	Australia	English	Chrome	86.0.4240.183	Macintosh Intel 10.15	Safari		5	1
1998758230.1607115407	20201204	Des Moines	United States	English	Chrome	86.0.4240.198	Windows 10	Chrome		5	1
2031226268.1607102245	20201204	Kharkiv	Ukraine	Russian	Opera	72.0.3815.400	Windows 10	Opera		4	1
2032810899.1607115902	20201204	Austin	United States	English	Firefox	82.0	Windows 10	Firefox		5	1
2039490199.1606995196	20201204	(not set)	Netherlands	English	Opera	72.0.3815.400	Windows 7	Opera		96	1
2039754430.1607110945	20201204	Kyiv	Ukraine	Russian	Chrome	71.0.3578.141	Android 10	Mi 10		7	1
2046988368.1607087804	20201204	Kyiv	Ukraine	Russian	Chrome	87.0.4280.66	Android 11	IN2020	8 Pro	4	1
2068355297.1606995142	20201204	(not set)	Netherlands	English	Opera	72.0.3815.400	Windows 7	Opera		351	7
2081493952.1607107703	20201204	Dublin	Ireland	English	Firefox	83.0	Linux	Firefox		5	1
2106084753.1607088297	20201204	Zaporizhzhia	Ukraine	Ukrainian	Samsung Internet	13.0	Android 9	SM-J730FM	Galaxy J7 Pro	6	1
2123374196.1607088964	20201204	Kharkiv	Ukraine	Russian	Safari	0605.1.15	iOS 13.4	iPhone		10	1
2135812031.1607117513	20201204	Munich	Germany	German	Chrome	86.0.4240.198	Windows 10	Chrome		8	1
2139494767.1607112242	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.66	Android 8.1.0	LG-M700	Q6	6	1
2141368623.1607111064	20201204	Kyiv	Ukraine	Ukrainian	Safari	14.0.1	Macintosh Intel 10.15	Safari		4	1
225520247.1607101434	20201204	Kyiv	Ukraine	Russian	Chrome	87.0.4280.86	Android 10	Mi 9T		5	1
225520247.1607101434	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.86	Android 10	Mi 9T		5	1

Ідентифікатор копії додатка	Дата	Місто	Країна	Мова	Веб-переглядач	Версія веб-переглядача	Операційна система та її версія	Модель мобільного пристрою	Пристрій	Кількість подій	Сеанси
24529994.1607103963	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.86	Android 10	ELE-L29	P30	4	1
248807390.1607099323	20201204	Kharkiv	Ukraine	Ukrainian	Chrome	87.0.4280.66	Android 10	IN2020	8 Pro	7	2
248807390.1607099323	20201204	Kyiv	Ukraine	Ukrainian	Chrome	87.0.4280.66	Android 10	IN2020	8 Pro	7	2
248965947.1607088249	20201204	Zaporizhzhia	Ukraine	Ukrainian	Chrome	86.0.4240.198	Windows 10	Chrome		8	1
267414720.1607114933	20201204	Austin	United States	English	Safari	13.01.2001	iOS 13.5.1	iPhone		4	1
299926010.1607098756	20201204	Leander	United States	English	Chrome	86.0.4240.198	Windows 10	Chrome		5	1
301121781.1607109480	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.66	Windows 10	Chrome		12	2
310775480.1607104123	20201204	Kharkiv	Ukraine	Russian	Firefox	83.0	Macintosh Intel 10.14	Safari		6	2
33864768.1607115307	20201204	Austin	United States	English	Chrome	87.0.4280.88	Windows 10	Chrome		5	1
350356004.1607086686	20201204	Kyiv	Ukraine	Russian	Safari	13.01.2001	iOS 13.5.1	iPhone		5	1
386533626.1607104292	20201204	Kharkiv	Ukraine	English	Safari	12.0.3	Macintosh Intel 10.14	Safari		4	1
39152030.1607112787	20201204	Kharkiv	Ukraine	Russian	Safari	14.0.1	iOS 14.2	iPhone		4	1
437329172.1607118633	20201204	Brisbane	Australia	English	Safari	14.0.1	iOS 14.2.1	iPhone		4	1
444798277.1607089866	20201204	Kharkiv	Ukraine	English	Safari	14.0.1	iOS 14.2	iPhone		4	1
467603027.1607112702	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.88	Windows 10	Chrome		5	1
477766545.1605962755	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.86	Android 10	Mi 9T		25	2
477766545.1605962755	20201204	Slough	United Kingdom	Russian	Chrome	87.0.4280.86	Android 10	Mi 9T		25	2
482782698.1607088080	20201204	Kharkiv	Ukraine	Russian	Edge	87.0.664.52	Windows 10	Edge		5	1
502286307.1607089830	20201204	Kharkiv	Ukraine	Russian	Firefox	77.0	Windows 10	Firefox		4	1
51305564.1607115804	20201204	Kyiv	Ukraine	English	Chrome	86.0.4240.198	Android 7.1.2	Redmi 4X		6	1
515051398.1607102804	20201204	Kharkiv	Ukraine	Russian	Firefox	83.0	Android 10	(not set)		7	1
52947145.1606974379	20201204	Kharkiv	Ukraine	Russian	Chrome	86.0.4240.198	Android 8.0	Pixel 2		300	3
52947145.1606974379	20201204	Kharkiv	Ukraine	Russian	Safari	11.0	iOS 11.0	iPad		8	0
52947145.1606974379	20201204	Kharkiv	Ukraine	Russian	Safari	10.0	iOS 10.3.1	iPhone		3	0
52947145.1606974379	20201204	Kharkiv	Ukraine	Russian	Safari	13.0.3	iOS 13.2.3	iPhone		3	0
538266522.1607102268	20201204	Kharkiv	Ukraine	Russian	Opera	72.0.3815.400	Windows 10	Opera		5	1
54996970.1607099125	20201204	Kharkiv	Ukraine	English	Safari	14.0.1	iOS 14.2	iPhone		31	1
556296286.1536018811	20201204	Kharkiv	Ukraine	Russian	Chrome	86.0.4240.198	Android 10	COL-L29	Honor 10	26	2
564278859.1607118359	20201204	Kharkiv	Ukraine	Russian	Chrome	86.0.4240.198	Android 11	Pixel 3a		6	1

Ідентифікатор копії додатка	Дата	Місто	Країна	Мова	Веб-переглядач	Версія веб-переглядача	Операційна система та її версія	Модель мобільного пристрою	Пристрій	Кількість подій	Сеанси
572481594.1607104697	20201204	Kyiv	Ukraine	Ukrainian	Chrome	87.0.4280.66	Android 10	IN2020	8 Pro	4	1
57462633.1547752531	20201204	Sumy	Ukraine	English	Chrome	71.0.3578.141	Android 7.0	Redmi Note 4		3	1
60858169.1607102189	20201204	Kharkiv	Ukraine	Russian	Firefox	83.0	Windows 10	Firefox		5	1
608734279.1607118659	20201204	Kyiv	Ukraine	Russian	Samsung Internet	13.0	Android 10	GM1910	7 Pro	4	1
616369222.1607088219	20201204	Zaporizhzhia	Ukraine	English	Firefox	83.0	Windows 8.1	Firefox		9	2
640698633.1607102819	20201204	Kharkiv	Ukraine	Russian	YaBrowser	20.11.1.88.00	Android 10	Mi 9T		12	1
671319554.1607102758	20201204	Kharkiv	Ukraine	Russian	UC Browser	13.3.2.1303	Android 10	Mi 9T		6	1
677853378.1580734752	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.66	Windows 10	Chrome		5	1
684470952.1604650691	20201204	Mariupol'	Ukraine	Russian	Chrome	86.0.4240.198	Android 10	EML-L29	P20	6	1
688596281.1607116683	20201204	Sumy	Ukraine	Russian	Chrome	87.0.4280.88	Windows 10	Chrome		4	1
723683170.1607102887	20201204	Kharkiv	Ukraine	Russian	Chrome	61.0.3163.128	Android 10	Mi 9T		4	1
753062036.1607099110	20201204	Kharkiv	Ukraine	English	Safari	14.0.1	iOS 14.2	iPhone		34	1
753062036.1607099110	20201204	Kyiv	Ukraine	English	Safari	14.0.1	iOS 14.2	iPhone		34	1
777777164.1607100800	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.88	Windows 10	Chrome		4	1
809908940.1607101697	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.88	Windows 10	Chrome		5	1
83332370.1607102341	20201204	Kharkiv	Ukraine	Russian	Safari	05.01.2007	Windows 8	(not set)		4	1
845312458.1607116436	20201204	Kyiv	Ukraine	English	Chrome	71.0.3578.141	Android 7.1.2	Redmi 4X		4	1
846256.1607086124	20201204	Kharkiv	Ukraine	English	Edge	18.19041	Windows 10	Edge		5	1
860405379.1607102851	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.86	Android 10	Mi 9T		10	1
865207203.1607090425	20201204	Kharkiv	Ukraine	English	Chrome	86.0.4240.111	Windows 10	Chrome		12	1
875509476.1607111209	20201204	Kyiv	Ukraine	Russian	Chrome	86.0.4240.198	Android 10	Mi 10		12	2
880973634.1607087007	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.66	Windows 10	Chrome		4	1
881306142.1574507972	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.86	Android 10	SM-A505FM	Galaxy A50	4	1
902764812.1607090306	20201204	Kharkiv	Ukraine	English	Chrome	87.0.4280.66	Android 8.1.0	Redmi 5 Plus		4	1
913003320.1592305255	20201204	Kharkiv	Ukraine	English	Chrome	87.0.4280.77	iOS 14.2	iPhone		13	1
925974708.1597397313	20201204	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.88	Windows 10	Chrome		23	1
948191450.1607114360	20201204	Kharkiv	Ukraine	English	Chrome	87.0.4280.88	Windows 7	Chrome		4	1
962304733.1607107964	20201204	Houston	United States	English	Safari	12.0	iOS 12.1.2	iPhone		4	1
98096643.1607102716	20201204	Kharkiv	Ukraine	Russian	Opera	60.3.3004.55692	Android 10	Mi 9T		6	1
1013471088.1551551777	20201205	Sumy	Ukraine	Russian	Chrome	87.0.4280.88	Windows 10	Chrome		6	2
1014246508.1607175788	20201205	Kharkiv	Ukraine	Russian	Safari	14.0.1	iOS 14.2	iPhone		4	1

Ідентифікатор копії додатка	Дата	Місто	Країна	Мова	Веб-переглядач	Версія веб-переглядача	Операційна система та її версія	Модель мобільного пристрою	Пристрій	Кількість подій	Сеанси
1021620600.1606841564	20201205	Kyiv	Ukraine	Ukrainian	Chrome	71.0.3578.141	Android 9	Redmi Note 5		7	1
1141127535.1607155954	20201205	Kharkiv	Ukraine	English	Chrome	87.0.4280.88	Windows 10	Chrome		5	1
1191788263.1541977410	20201205	Kharkiv	Ukraine	Russian	Chrome	86.0.4240.198	Windows 7	Chrome		168	7
1271457051.1607127433	20201205	Kharkiv	Ukraine	English	Chrome	86.0.4240.198	Windows 10	Chrome		5	1
1287751386.1607161483	20201205	Yalta	Ukraine	English	Chrome	87.0.4280.88	Windows 7	Chrome		4	1
1388046251.1607173199	20201205	Kyiv	Ukraine	English	Safari	14.0.1	iOS 14.2	iPhone		7	1
1452327580.1607161280	20201205	(not set)	Netherlands	Russian	Opera	72.0.3815.400	Windows 7	Opera		4	1
1494145258.1607177544	20201205	Zaporizhzhia	Ukraine	Russian	Firefox	83.0	Windows 10	Firefox		5	1
1619823079.1607130563	20201205	Kharkiv	Ukraine	Russian	Chrome	83.0.4103.106	Android 10	COL-L29	Honor 10	13	3
1619823079.1607130563	20201205	Kyiv	Ukraine	Russian	Chrome	83.0.4103.106	Android 10	COL-L29	Honor 10	13	3
1641809714.1607191478	20201205	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.66	Linux	Chrome		5	1
1648716722.1599503279	20201205	Kharkiv	Ukraine	Russian	Chrome	86.0.4240.198	Windows 10	Chrome		4	1
1661108203.1606973814	20201205	Kharkiv	Ukraine	Russian	Chrome	86.0.4240.198	Android 6.0.1	Moto G (4)	Moto G4	8	0
1713732424.1607156030	20201205	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.86	Android 10	MAR-LX1A	P30 Lite	4	1
1721210915.1607176933	20201205	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.88	Windows 10	Chrome		6	2
1731038829.1607125078	20201205	Kharkiv	Ukraine	Russian	Safari	14.0	iOS 14.0	iPhone		5	1
1796743927.1592412263	20201205	Presov	Slovakia	Russian	Chrome	87.0.4280.66	Android 9	Mi Note 3		4	1
1801383376.1607086702	20201205	Kyiv	Ukraine	Russian	Chrome	85.0.4183.109	iOS 13.5	iPhone		3	1
1813493933.1607119932	20201205	Kharkiv	Ukraine	English	Safari	14.0	iOS 14.0.1	iPhone		6	2
1876145276.1607177189	20201205	Kharkiv	Ukraine	Russian	Chrome	85.0.4183.127	Android 6.0	Redmi Note 4		4	1
1880935632.1589558139	20201205	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.77	iOS 14.2	iPhone		6	2
1937022894.1567397044	20201205	Kharkiv	Ukraine	English	Chrome	87.0.4280.77	iOS 14.2	iPhone		7	2
1954504288.1607085965	20201205	Kharkiv	Ukraine	English	Chrome	87.0.4280.66	Windows 10	Chrome		11	2
2041264396.1605020685	20201205	Yalta	Ukraine	English	Firefox	60.0	Windows 7	Firefox		6	2
2123374196.1607088964	20201205	Kharkiv	Ukraine	Russian	Safari	0605.1.15	iOS 13.4	iPhone		14	2
225226122.1607174684	20201205	Kyiv	Ukraine	Russian	Chrome	87.0.4280.86	Android 9	ANE-LX1	P20 Lite	6	1
265857836.1607173535	20201205	Kharkiv	Ukraine	English	Chrome	86.0.4240.198	Macintosh Intel 11.0	Safari		3	1
273278261.1607127461	20201205	Kharkiv	Ukraine	Russian	Edge	87.0.664.52	Windows 10	Edge		4	1
394139782.1607155866	20201205	Kharkiv	Ukraine	English	Chrome	87.0.4280.88	Windows 10	Chrome		5	1
417742564.1590673654	20201205	Kharkiv	Ukraine	Russian	Chrome	86.0.4240.198	Windows 10	Chrome		5	1
50972342.1605962324	20201205	Kharkiv	Ukraine	Russian	Chrome	87.0.4280.66	Windows 10	Chrome		5	1
52947145.1606974379	20201205	Kharkiv	Ukraine	Russian	Chrome	86.0.4240.198	Android 6.0	Nexus 5		721	8

Ідентифікатор копії додатка	Дата	Місто	Країна	Мова	Веб-переглядач	Версія веб-переглядача	Операційна система та її версія	Модель мобільного пристрою	Пристрій	Кількість подій	Сеанси
536903440.1607150898	20201205	Kyiv	Ukraine	Russian	Chrome	87.0.4280.86	Android 8.0.0	PRA-LA1	P8 Lite (2017)	6	1
556296286.1536018811	20201205	Kharkiv	Ukraine	Russian	Chrome	86.0.4240.198	Android 10	COL-L29	Honor 10	35	3
594489475.1607150977	20201205	Kyiv	Ukraine	Russian	Chrome	87.0.4280.86	Android 8.0.0	PRA-LA1	P8 Lite (2017)	4	1
731760463.1607156130	20201205	Kharkiv	Ukraine	English	Chrome	87.0.4280.86	Android 10	MAR-LX1A	P30 Lite	4	1
73238419.1571073474	20201205	Kharkiv	Ukraine	Russian	Chrome	86.0.4240.183	Macintosh Intel 10.14	Safari		6	2
73957761.1606992721	20201205	Kyiv	Ukraine	Russian	Firefox	83.0	Windows 7	Firefox		5	1
775790148.1607177000	20201205	Kharkiv	Ukraine	Russian	Safari	12.01.2001	Macintosh Intel 10.14	Safari		4	1
784283595.1607173140	20201205	Kyiv	Ukraine	English	Safari	14.0.1	iOS 14.2	iPhone		8	1
794979073.1604910793	20201205	Solonyt sivka	Ukraine	English	Chrome	86.0.4240.198	Android 10	Mi 9 Lite		5	1
832937423.1607190909	20201205	Kharkiv	Ukraine	English	Chrome	87.0.4280.66	Windows 10	Chrome		10	2
877503146.1607125171	20201205	Kharkiv	Ukraine	English	Safari	14.0	Macintosh Intel 10.15	Safari		4	1

ДОДАТОК В

Експериментальні дані

#	Platform	Vendor	Screen	Window	gaID	fpID	Version
0 1	Win32	Google Inc.	1920x1 080x24	1920x1 040x24	1191788263.1 541977410	6595 9D93	5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36
0 1	Win32	Google Inc.	1920x1 080x24	1920x1 040x24	1872751035.1 601081563	6595 9D93	5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36 OPR/72.0.3815.400
0 1	Linux armv8l	Google Inc.	424x89 5x24	424x89 5x24	1619823079.1 607130563	FA38 52FF	5.0 (Linux; Android 10; COL-L29; HMSCore 5.0.1.307; GMSCore 20.42.17) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/83.0.4103.106 HuaweiBrowser/11.0.3.304 Mobile Safari/537.36
0 1	Linux armv8l	Google Inc.	424x89 5x24	424x89 5x24	556296286.15 36018811	AA6F 9BD5	5.0 (Linux; Android 10; COL-L29) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Mobile Safari/537.36
0 1	Win32		1477x8 31x24	1477x8 00x24	1280424927.1 367851468	47B2 B7BA	5.0 (Windows NT 6.1; WOW64; Trident/7.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; .NET4.0C; .NET4.0E; rv:11.0) like Gecko
0 2	Win32	Google Inc.	1920x1 080x24	1920x1 040x24	nothing	AE08 E722	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36 OPR/72.0.3815.400
0 2	Win32	Google Inc.	1920x1 080x24	1920x1 040x24	nothing	7FAA BA10	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.66 Safari/537.36
0 2	Win32		1920x1 080x24	1920x1 040x24	nothing	74BD 02D3	5.0 (Windows NT 10.0; WOW64; Trident/7.0; .NET4.0C; .NET4.0E; rv:11.0) like Gecko
0 2	Win32		1920x1 080x24	1920x1 040x24	nothing	C753 FECB	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.102 Safari/537.36 Edge/18.19041
0 2	iPhone	Apple Computer, Inc.	375x81 2x32	375x81 2x32	913003320.15 923	4669 BD92	5.0 (iPhone; CPU iPhone OS 14_2 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/87.0.4280.77 Mobile/15E148 Safari/604.1
0 2	iPhone	Apple Computer, Inc.	375x81 2x32	375x81 2x32	913003320.15 92305255	4669 BD92	5.0 (iPhone; CPU iPhone OS 14_2 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/87.0.4280.77 Mobile/15E148 Safari/604.1
0 2	iPhone	Apple Computer, Inc.	375x81 2x32	375x81 2x32	54996970.160 7099125	1473F 34E	5.0 (iPhone; CPU iPhone OS 14_2 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/14.0.1 Mobile/15E148 Safari/604.1
0 2	iPhone	Apple Computer, Inc.	375x81 2x32	375x81 2x32	nothing	1473F 34E	5.0 (iPhone; CPU iPhone OS 14_2 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/14.0.1 Mobile/15E148 Safari/604.1
0 2	iPhone	Apple Computer, Inc.	375x81 2x32	375x81 2x32	753062036.16 07099110	1473F 34E	5.0 (iPhone; CPU iPhone OS 14_2 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/14.0.1 Mobile/15E148 Safari/604.1
0 3	Win32	Google Inc.	1536x8 64x24	1536x8 34x24	925974708.15 97397313	A764 1C4B	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36
0 3	Win32	Google Inc.	864x15 36x24	864x15 06x24	925974708.15 97397313	A764 1C4B	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36

#	Platform	Vendor	Screen	Window	gaID	fpID	Version
03	Win32	Google Inc.	864x1536x24	864x1506x24	925974708.1597397313	A7641C4B	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36
03	Win32	Google Inc.	1536x864x24	1536x834x24	nothing	A7641C4B	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.67 Safari/537.36 Edg/87.0.664.52
03	Win32	Google Inc.	1536x864x24	1536x834x24	925974708.1597397313	A7641C4B	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36
034	Linux aarch64	Google Inc.	851x393x24	851x393x24	1445935935.16026	73D7E572	5.0 (Linux; U; Android 10; ru-ru; Mi 9T Build/QKQ1.190825.002) AppleWebKit/537.36 (KHTML, like Gecko) Version/4.0 Chrome/71.0.3578.141 Mobile Safari/537.36 XiaoMi/MiuiBrowser/12.6.6-gn
034	Linux aarch64	Google Inc.	851x393x24	851x393x24	1445935935.16026	73D7E572	5.0 (Linux; U; Android 10; ru-ru; Mi 9T Build/QKQ1.190825.002) AppleWebKit/537.36 (KHTML, like Gecko) Version/4.0 Chrome/71.0.3578.141 Mobile Safari/537.36 XiaoMi/MiuiBrowser/12.6.6-gn
034	Linux aarch64	Google Inc.	393x851x24	393x851x24	1445935935.16026	73D7E572	5.0 (Linux; U; Android 10; ru-ru; Mi 9T Build/QKQ1.190825.002) AppleWebKit/537.36 (KHTML, like Gecko) Version/4.0 Chrome/71.0.3578.141 Mobile Safari/537.36 XiaoMi/MiuiBrowser/12.6.6-gn
03	Linux x86_64	Google Inc.	850x392x24	850x392x24	1428803790.1607	7F554ECF	5.0 (Linux; Android 10; Mi 9T Build/QKQ1.190825.002; ru-ru) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/69.0.3497.100 Mobile Safari/537.36 Puffin/8.5.0.42207AP
03	Linux x86_64	Google Inc.	850x392x24	850x392x24	1428803790.1607	7F554ECF	5.0 (Linux; Android 10; Mi 9T Build/QKQ1.190825.002; ru-ru) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/69.0.3497.100 Mobile Safari/537.36 Puffin/8.5.0.42207AP
03	Linux x86_64	Google Inc.	392x850x24	392x850x24	nothing	7F554ECF	5.0 (Linux; Android 10; Mi 9T Build/QKQ1.190825.002; ru-ru) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/69.0.3497.100 Mobile Safari/537.36 Puffin/8.5.0.42207AP
03	Linux armv8l	Google Inc.	393x851x24	393x851x24	477766545.160596	2BDEA1FC	5.0 (Linux; Android 10; Mi 9T) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.86 Mobile Safari/537.36
03	Linux armv8l	Google Inc.	851x393x24	851x393x24	477766545.160596	2BDEA1FC	5.0 (Linux; Android 10; Mi 9T) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.86 Mobile Safari/537.36
03	Linux armv8l	Google Inc.	851x393x24	851x393x24	477766545.160596	4CFDF9E1	5.0 (Linux; Android 10; Mi 9T) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.86 Mobile Safari/537.36
03	Linux armv8l	Google Inc.	393x851x24	393x851x24	477766545.160596	4CFDF9E1	5.0 (Linux; Android 10; Mi 9T) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.86 Mobile Safari/537.36
03	Linux armv8l	Google Inc.	851x393x24	851x393x24	477766545.160596	4CFDF9E1	5.0 (Linux; Android 10; Mi 9T) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.86 Mobile Safari/537.36
03	Linux armv8l	Google Inc.	851x393x24	851x393x24	477766545.160596	4CFDF9E1	5.0 (Linux; Android 10; Mi 9T) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.86 Mobile Safari/537.36
03	Linux armv8l	Google Inc.	393x851x24	393x851x24	477766545.160596	4CFDF9E1	5.0 (Linux; Android 10; Mi 9T) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.86 Mobile Safari/537.36
03	Linux armv8l	Google Inc.	393x851x24	393x851x24	225520247.16071	2BDEA1FC	5.0 (Linux; Android 10; Mi 9T) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.86 Mobile Safari/537.36

#	Platform	Vendor	Screen	Window	gaID	fpID	Version
03	Win32	Google Inc.	1536x864x24	1536x834x24	nothing	A7641C4B	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36
04	iPhone	Apple Computer, Inc.	350x568x32	350x568x32	2123374196.1607088964	CC222966	5.0 (iPhone; CPU iPhone OS 13_4 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) FxiOS/13.1b11805 Mobile/15E148 Safari/605.1.15
04	MacIntel		1440x900x24	1440x877x24	nothing	C6D6339A	5.0 (Macintosh)
04	MacIntel		1440x900x24	1440x877x24	nothing	C6D6339A	5.0 (Macintosh)
05	Linux armv8l	Google Inc.	360x780x24	360x780x24	nothing	BAD3E6C	5.0 (Linux; Android 10; ELE-L29) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.86 Mobile Safari/537.36
05	Win32	Google Inc.	1280x720x24	1280x682x24	nothing	BA6F3451	5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36
05	Win32	Google Inc.	1280x720x24	1280x682x24	nothing	BA6F3451	5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36
06	iPhone	Apple Computer, Inc.	375x667x32	375x667x32	nothing	2C76EB1D	5.0 (iPhone; CPU iPhone OS 13_5 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/85.0.4183.109 Mobile/15E148 Safari/604.1
06	Win32	Google Inc.	1920x1080x24	1920x1050x24	nothing	B9FCAF0	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36
06	Win32	Google Inc.	1920x1080x24	1920x1050x24	nothing	42E0F1C6	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.67 Safari/537.36 Edg/87.0.664.52
06	iPhone	Apple Computer, Inc.	375x667x32	375x667x32	350356004.1607086686	2C76EB1D	5.0 (iPhone; CPU iPhone OS 13_5_1 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/13.1.1 Mobile/15E148 Safari/604.1
07	Linux armv8l	Google Inc.	393x830x24	393x830x24	1689492983.1607	3E823C88	5.0 (Linux; Android 11; Pixel 4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Mobile Safari/537.36
07	Win32		1280x720x24	1280x680x24	nothing	21E7D9	5.0 (Windows)
07	Win32	Google Inc.	1280x720x24	1280x680x24	nothing	B9FCAF0	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/77.0.3865.90 Safari/537.36
08	Win32	Google Inc.	2560x1440x24	2560x1400x24	nothing	9D35FE62	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.66 Safari/537.36
08	Win32	Google Inc.	2560x1440x24	2560x1400x24	nothing	9D35FE62	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.66 Safari/537.36
08	Linux aarch64		396x792x24	396x792x24	nothing	C61230EC	5.0 (Android 9)

#	Platform	Vendor	Screen	Window	gaID	fpID	Version
08	Linux aarch64		396x792x24	396x792x24	nothing	C61230EC	5.0 (Android 9)
09	Linux armv8l	Google Inc.	385x845x24	385x845x24	nothing	85EC5BED	5.0 (Linux; Android 11; IN2020) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.66 Mobile Safari/537.36
09	Win32	Google Inc.	1920x1080x24	1920x1040x24	nothing	B9FC AF0	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.111 Safari/537.36
09	Win32	Google Inc.	1920x1080x24	1920x1040x24	nothing	B9FC AF0	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.111 Safari/537.36
10	Linux armv7l		414x808x24	414x808x24	nothing	8BDE7F40	5.0 (Android 8.1)
10	Linux armv8l	Google Inc.	412x892x24	412x892x24	1413985239.15971	18665BCF	5.0 (Linux; Android 10; SN) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.3945.116 Mobile Safari/537.36
11	Win32	Google Inc.	1920x1080x24	1858x1080x24	nothing	7D4C A8B	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36
11	Win32		1920x1080x24	1858x1080x24	nothing	997B A88A	5.0 (Windows)
11	Win32	Google Inc.	1920x1080x24	1858x1080x24	nothing	7D4C A8B	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36
11	iPhone	Apple Computer, Inc.	414x736x32	414x736x32	nothing	425B07BC	5.0 (iPhone; CPU iPhone OS 14_2 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/14.0.1 Mobile/15E148 Safari/604.1
12	Linux armv8l	Google Inc.	393x786x24	393x786x24	nothing	962E7B40	5.0 (Linux; Android 8.1.0; Redmi 5 Plus) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.66 Mobile Safari/537.36
13	Win32	Google Inc.	1638x959x24	1638x919x24	nothing	64EF A1E9	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36
13	Win32	Google Inc.	1920x1080x24	1920x1040x24	nothing	E346 BA65	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36
13	Win32	Google Inc.	1920x1080x24	1920x1080x24	nothing	42E0F1C6	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36
14	MacIntel	Google Inc.	1680x1050x30	1680x929x30	nothing	734D FD58	5.0 (Macintosh; Intel Mac OS X 11_0_0) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36
14	Linux armv8l	Google Inc.	412x906x24	412x906x24	nothing	4CFD F9E1	5.0 (Linux; Android 10; IN2020) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.66 Mobile Safari/537.36
14	Linux armv8l	Google Inc.	412x906x24	412x906x24	nothing	2BDE A1FC	5.0 (Linux; Android 10; IN2020) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.66 Mobile Safari/537.36
05	Win32	Google Inc.	1280x720x24	1280x82x24	nothing	BA6F3451	5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36

#	Platform	Vendor	Screen	Window	gaID	fpID	Version
15	Linux x86_64		1920x1080x24	1853x1053x24	nothing	1F0F9F9C	5.0 (X11)
06	Win32	Google Inc.	1920x1080x24	1920x1050x24	nothing	B9FC AF0	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36
06	Win32	Google Inc.	1920x1080x24	1920x1050x24	nothing	42E0F1C6	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.67 Safari/537.36 Edg/87.0.664.52
06	iPhone	Apple Computer, Inc.	375x667x32	375x667x32	350356004.1607086686	2C76 EB1D	5.0 (iPhone; CPU iPhone OS 13_5_1 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/13.1.1 Mobile/15E148 Safari/604.1
06	iPhone	Apple Computer, Inc.	375x667x32	375x667x32	nothing	2C76 EB1D	5.0 (iPhone; CPU iPhone OS 13_5 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/85.0.4183.109 Mobile/15E148 Safari/604.1
17	Win32	Google Inc.	1536x864x24	1536x824x24	nothing	B9FC AF0	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36
07	Linux armv8l	Google Inc.	393x830x24	393x830x24	1689492983.1607086997	3E82 3C88	5.0 (Linux; Android 11; Pixel 4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Mobile Safari/537.36
07	Win32	Google Inc.	1280x720x24	1280x680x24	nothing	B9FC AF0	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/77.0.3865.90 Safari/537.36
18	Linux armv8l	Google Inc.	360x748x24	360x748x24	684470952.160465	727A F0EE	5.0 (Linux; Android 10; EML-L29) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Mobile Safari/537.36
18	Linux armv8l	Google Inc.	360x748x24	360x748x24	nothing	AA6F 9BD5	5.0 (Linux; Android 10; EML-L29) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Mobile Safari/537.36
19	Linux armv8l	Google Inc.	360x740x24	360x740x24	nothing	CDBA 017F	5.0 (Linux; Android 9; SM-G950U) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Mobile Safari/537.36
19	Win32	Google Inc.	1536x864x24	1536x824x24	nothing	B9FC AF0	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36
19	Win32	Google Inc.	1536x864x24	1536x824x24	nothing	B9FC AF0	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36
20	Win32		1366x768x24	1366x728x24	nothing	4282F DB9	5.0 (Windows)
21	Linux armv8l	Google Inc.	360x771x24	360x771x24	nothing	9EF86 EC	5.0 (Linux; Android 10; MAR-LX1A) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.86 Mobile Safari/537.36
21	Linux armv8l	Google Inc.	360x771x24	360x771x24	nothing	BAD3 E6C	5.0 (Linux; Android 10; MAR-LX1A) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.86 Mobile Safari/537.36
21	Win32	Google Inc.	1920x1080x24	1920x1050x24	nothing	9D35 FE62	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36
21	Win32	Google Inc.	1920x1080x24	1920x1050x24	nothing	9D35 FE62	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36

#	Platform	Vendor	Screen	Window	gaID	fpID	Version
2 2	Linux armv8l	Google Inc.	450x80 0x24	450x80 0x24	1796743927.1 592412263	962E 7B40	5.0 (Linux; Android 9; Mi Note 3) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.66 Mobile Safari/537.36
2 2	Win32	Google Inc.	1536x8 64x24	1468x8 64x24	nothing	B9FC AF0	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/80.0.3987.163 Safari/537.36
2 3	Linux armv8l	Google Inc.	360x76 0x24	360x76 0x24	1749194371.1 607110435	6B4E 2E9F	5.0 (Linux; Android 9; ANE-LX1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.66 Mobile Safari/537.36
2 4	Linux armv8l	Google Inc.	393x85 1x24	393x85 1x24	794979073.16 0491	2B92 E7A6	5.0 (Linux; Android 10; Mi 9 Lite) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Mobile Safari/537.36
2 5	Win32	Google Inc.	1366x7 68x24	1366x7 28x24	nothing	E9EC DOC6	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.66 Safari/537.36
2 5	Win32	Google Inc.	1920x1 080x24	1920x1 040x24	301121781.16 07109480	E9EC DOC6	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.66 Safari/537.36
2 5	Linux armv8l	Google Inc.	360x64 0x32	360x64 0x32	1328669945.1 607109586	C9FE C74C	5.0 (Linux; U; Android 6.0; zh-CN; MZ-M5s Build/MRA58K) AppleWebKit/537.36 (KHTML, like Gecko) Version/4.0 Chrome/57.0.2987.108 MZBrowser/8.4.110-2020111017 UWS/2.15.0.4 Mobile Safari/537.36
2 6	iPhone	Apple Computer, Inc.	375x66 7x32	375x66 7x32	1731038829.1 607125078	425B 07BC	5.0 (iPhone; CPU iPhone OS 14_0 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/14.0 Mobile/15E148 Safari/604.1
2 6	MacIntel	Apple Computer, Inc.	1024x1 366x32	1366x1 024x32	nothing	C250 3854	5.0 (Macintosh; Intel Mac OS X 10_15_6) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/14.0 Safari/605.1.15
2 7	Linux armv7l	Google Inc.	360x64 0x24	360x64 0x24	nothing	BC9B C081	5.0 (Linux; Android 8.0.0; PRA-LA1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.86 Mobile Safari/537.36
2 7	Linux armv7l	Google Inc.	360x64 0x24	360x64 0x24	nothing	6D21 39CE	5.0 (Linux; Android 8.0.0; PRA-LA1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.86 Mobile Safari/537.36
2 7	Win32	Google Inc.	1366x7 68x24	1366x7 68x24	nothing	F8A8 5043	5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/83.0.4103.97 Safari/537.36
2 7	Win32	Google Inc.	1366x7 68x24	1366x7 68x24	nothing	F8A8 5043	5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/83.0.4103.97 Safari/537.36
2 7	Win32	Google Inc.	1366x7 68x24	1366x7 68x24	nothing	F8A8 5043	5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/83.0.4103.97 Safari/537.36
2 8	Linux armv8l	Google Inc.	393x80 8x24	393x80 8x24	564278859.16 07118359	3E82 3C88	5.0 (Linux; Android 11; Pixel 3a) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Mobile Safari/537.36
2 8	Linux x86_64	Google Inc.	1920x1 080x24	1848x1 053x24	nothing	4C0C 45EC	5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.66 Safari/537.36
2 9	Win32	Google Inc.	1920x1 080x24	1920x1 040x24	677853378.15 80734752	42E0F 1C6	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.66 Safari/537.36

#	Platform	Vendor	Screen	Window	gaID	fpID	Version
29	iPhone	Apple Computer, Inc.	414x89 6x32	414x89 6x32	1880935632.1 589558139	C250 3854	5.0 (iPhone; CPU iPhone OS 14_2 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/87.0.4280.77 Mobile/15E148 Safari/604.1
30	Linux armv8l	Google Inc.	360x64 0x24	360x64 0x24	nothing	187F DF0C	5.0 (Linux; Android 7.1.1; ASUS_X00LDA) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Mobile Safari/537.36
30	Linux armv8l	Google Inc.	360x64 0x24	360x64 0x24	1534916846.1 548	3B8F 4E6A	5.0 (Linux; Android 7.1.1; ASUS_X00LDA) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Mobile Safari/537.36
30	Linux armv8l	Google Inc.	360x64 0x24	360x64 0x24	1534916846.1 548	3B8F 4E6A	5.0 (Linux; Android 7.1.1; ASUS_X00LDA) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Mobile Safari/537.36
31	Linux armv8l	Google Inc.	393x85 1x24	393x85 1x24	875509476.16 07111209	D848 779B	5.0 (Linux; Android 10; Mi 10) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Mobile Safari/537.36
31	Linux armv8l	Google Inc.	393x85 1x24	393x85 1x24	nothing	D848 779B	5.0 (Linux; Android 10; Mi 10) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Mobile Safari/537.36
32	iPhone	Apple Computer, Inc.	414x89 6x32	414x89 6x32	1663323080.1 607110958	C250 3854	5.0 (iPhone; CPU iPhone OS 14_2 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/14.0.1 Mobile/15E148 Safari/604.1
32	iPhone	Apple Computer, Inc.	414x89 6x32	414x89 6x32	nothing	C250 3854	5.0 (iPhone; CPU iPhone OS 14_2 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/14.0.1 Mobile/15E148 Safari/604.1
32	MacIntel	Apple Computer, Inc.	768x10 24x32	1024x7 68x32	nothing	3C65 2080	5.0 (Macintosh; Intel Mac OS X 10_15_6) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/14.0.1 Safari/605.1.15
33	Win32	Google Inc.	2560x1 440x24	2560x1 400x24	1424391584.1 606839313	7D4C A8B	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36
33	Linux armv7l	Google Inc.	360x72 0x24	360x72 0x24	1547951770.1 5988	2FB7 5E66	5.0 (Linux; U; Android 9; ru-ru; Redmi 6A Build/PPR1.180610.011) AppleWebKit/537.36 (KHTML, like Gecko) Version/4.0 Chrome/71.0.3578.141 Mobile Safari/537.36 XiaoMi/MiuiBrowser/12.5.2-go
34	Linux armv7l	Google Inc.	360x72 0x24	360x72 0x24	2139494767.1 607112242	962E 7B40	5.0 (Linux; Android 8.1.0; LG-M700) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.66 Mobile Safari/537.36
34	Win32	Google Inc.	1920x1 080x24	1920x1 040x24	1815949238.1 601323291	7D4C A8B	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36 OPR/72.0.3815.400
35	Win32	Google Inc.	1418x7 98x24	1418x7 59x24	nothing	A764 1C4B	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36
35	Win32		1408x7 92x24	1408x7 53x24	nothing	8838 625	5.0 (Windows)
35	iPhone	Apple Computer, Inc.	375x81 2x32	375x81 2x32	nothing	C250 3854	5.0 (iPhone; CPU iPhone OS 14_2 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/14.0.1 Mobile/15E148 Safari/604.1

#	Platform	Vendor	Screen	Window	gaID	fpID	Version
36	Win32	Google Inc.	2195x1235x24	2195x195x24	nothing	B9FC AF0	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36
36	Win32	Google Inc.	1920x1080x24	1920x1040x24	nothing	B9FC AF0	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36
36	iPhone	Apple Computer, Inc.	414x896x32	414x896x32	nothing	2C76 EB1D	5.0 (iPhone; CPU iPhone OS 13_5_1 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/13.1.1 Mobile/15E148 Safari/604.1
37	Win32	Google Inc.	1920x1080x24	1920x1040x24	50972342.160 5962324	9D35 FE62	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.66 Safari/537.36
37	iPhone	Apple Computer, Inc.	414x896x32	414x896x32	nothing	C250 3854	5.0 (iPhone; CPU iPhone OS 14_2 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/14.0.1 Mobile/15E148 Safari/604.1
37	MacIntel	Google Inc.	1440x900x24	1440x900x24	73238419.157 1073474	48C0 F26B	5.0 (Macintosh; Intel Mac OS X 10_14_5) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.183 Safari/537.36
38	Win32	Google Inc.	1366x768x24	1366x728x24	nothing	4236 D67C	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36
38	Linux armv8l	Google Inc.	360x640x24	360x640x24	nothing	F5D5 D319	5.0 (Linux; Android 6.0; Redmi Note 4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/85.0.4183.127 Mobile Safari/537.36
38	Win32	Google Inc.	1920x1080x24	1920x1040x24	nothing	50B8 2C5C	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36
39	MacIntel	Google Inc.	2560x1440x24	2560x1374x24	1966650927.1 583782392	A6E9 50BD	5.0 (Macintosh; Intel Mac OS X 11_0_0) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36
39	iPhone	Apple Computer, Inc.	414x896x32	414x896x32	1937022894.1 567397044	C250 3854	5.0 (iPhone; CPU iPhone OS 14_2 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/87.0.4280.77 Mobile/15E148 Safari/604.1
39	MacIntel	Apple Computer, Inc.	2560x1440x24	2560x1374x24	nothing	4AF4 EB86	5.0 (Macintosh; Intel Mac OS X 10_15_6) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/14.0.1 Safari/605.1.15
41	Win32	Google Inc.	1920x1080x24	1920x1040x24	nothing	F72A 26BD	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36
41	Win32	Google Inc.	1920x1080x24	1920x1040x24	nothing	A764 1C4B	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.67 Safari/537.36 Edg/87.0.664.55
41	Linux armv8l	Google Inc.	412x892x24	412x892x24	nothing	2B92 E7A6	5.0 (Linux; Android 10; GM1900) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Mobile Safari/537.36
42	Linux armv8l	Google Inc.	360x640x24	360x640x24	51305564.160 7115804	F1287 F1A	5.0 (Linux; Android 7.1.2; Redmi 4X) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Mobile Safari/537.36
42	Linux armv8l	Google Inc.	360x640x24	360x640x24	nothing	E5F3 AC1A	5.0 (Linux; U; Android 7.1.2; en-gb; Redmi 4X Build/N2G47H) AppleWebKit/537.36 (KHTML, like Gecko) Version/4.0 Chrome/71.0.3578.141 Mobile Safari/537.36 XiaoMi/MiuiBrowser/12.5.2-go

#	Platform	Vendor	Screen	Window	gaID	fpID	Version
43	Win32	Google Inc.	1920x1080x24	1920x1050x24	417742564.1590673654	7D4CA8B	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36
43	Linux armv8l	Google Inc.	393x786x24	393x786x24	1021620600.1606841564	2FB75E66	5.0 (Linux; U; Android 9; uk-ua; Redmi Note 5 Build/PKQ1.180904.001) AppleWebKit/537.36 (KHTML, like Gecko) Version/4.0 Chrome/71.0.3578.141 Mobile Safari/537.36 Xiaomi/MiuiBrowser/12.5.2-go
44	Win32	Google Inc.	1920x1080x24	1920x1040x24	1013471088.1551551777	E9ECDOC6	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36
44	Linux armv8l	Google Inc.	360x640x24	360x640x24	57462633.1547752531	E5F3AC1A	5.0 (Linux; U; Android 7.0; en-us; Redmi Note 4 Build/NRD90M) AppleWebKit/537.36 (KHTML, like Gecko) Version/4.0 Chrome/71.0.3578.141 Mobile Safari/537.36 Xiaomi/MiuiBrowser/10.9.9-g
44	Win32	Google Inc.	1920x1080x24	1920x1040x24	nothing	E9ECDOC6	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36
44	Win32		1920x1080x24	1920x1040x24	nothing	4282FDB9	5.0 (Windows)
44	Win32	Google Inc.	1920x1080x24	1920x1040x24	1372173739.1567893189	2CB86C62	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36 OPR/72.0.3815.400
44	Win32	Google Inc.	1920x1080x24	1920x1040x24	nothing	E9ECDOC6	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.67 Safari/537.36 Edg/87.0.664.52
45	MacIntel	Google Inc.	1680x1050x24	1680x1025x24	265857836.1607173535	1B94E6BA	5.0 (Macintosh; Intel Mac OS X 11_0_0) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36
45	iPhone	Apple Computer, Inc.	375x667x32	375x667x32	784283595.1607173140	4669BD92	5.0 (iPhone; CPU iPhone OS 14_2 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/14.0.1 Mobile/15E148 Safari/604.1
46	Win32		1536x864x24	1536x824x24	157673876.157564	253F92D9	5.0 (Windows)
46	Linux armv8l	Google Inc.	360x748x24	360x748x24	nothing	C81AB0C0	5.0 (Linux; Android 8.1.0; ARE-AL00) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.66 Mobile Safari/537.36
47	Linux armv8l	Google Inc.	412x892x24	412x892x24	1209199027.1607118618	2B92E7A6	5.0 (Linux; Android 10; GM1910) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Mobile Safari/537.36
47	Linux armv8l	Google Inc.	412x892x24	412x892x24	nothing	D848779B	5.0 (Linux; Android 10; GM1910) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Mobile Safari/537.36
48	Win32	Google Inc.	1920x1080x24	1920x1040x24	1037537196.1580246722	9D35FE62	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36
48	Linux armv8l	Google Inc.	412x892x24	412x892x24	881306142.1574507972	4DC2CE39	5.0 (Linux; Android 10; SM-A505FM) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.86 Mobile Safari/537.36
49	Linux armv8l	Google Inc.	360x740x24	360x740x24	nothing	9D8640B5	5.0 (Linux; Android 9; SM-G950F) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Mobile Safari/537.36
49	Win32	Google Inc.	1366x768x24	1366x728x24	nothing	B9FCAF0	5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36

#	Platform	Vendor	Screen	Window	gaID	fpID	Version
50	Linux armv8l	Google Inc.	360x780x24	360x780x24	nothing	30D2CE29	5.0 (Linux; Android 9; INE-LX1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Mobile Safari/537.36
50	Win32		1536x864x24	1536x824x24	nothing	BF53730E	5.0 (Windows)
51	Win32		1366x768x24	1366x740x24	73957761.1606992	C63D5DE2	5.0 (Windows)
51	Win32	Google Inc.	1366x768x24	1366x740x24	nothing	3CFF42A7	5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36 OPR/72.0.3815.400
52	MacIntel	Google Inc.	3440x1440x24	3440x1417x24	nothing	734DFD58	5.0 (Macintosh; Intel Mac OS X 10_15_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.183 Safari/537.36
52	iPhone	Apple Computer, Inc.	414x896x32	414x896x32	nothing	DEB9721D	5.0 (iPhone; CPU iPhone OS 14_2_1 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/14.0.1 Mobile/15E148 Safari/604.1