I don't have an accurate number to prove but based on my past experiences, online marketing nowadays seems more economical and efficient than conventional marketing schemes, such as TV commercials. Imagine if I have a website (I do have one) or an online store, with the help of digital (web traffic) analytics I can answer three questions in no time:

1. Who are my visitors (and how many)? (Acquisition)

2. What do they do on my website (and how many)? (Behaviour)

3. Who became my customers eventually (and the ratio)? (Conversion)
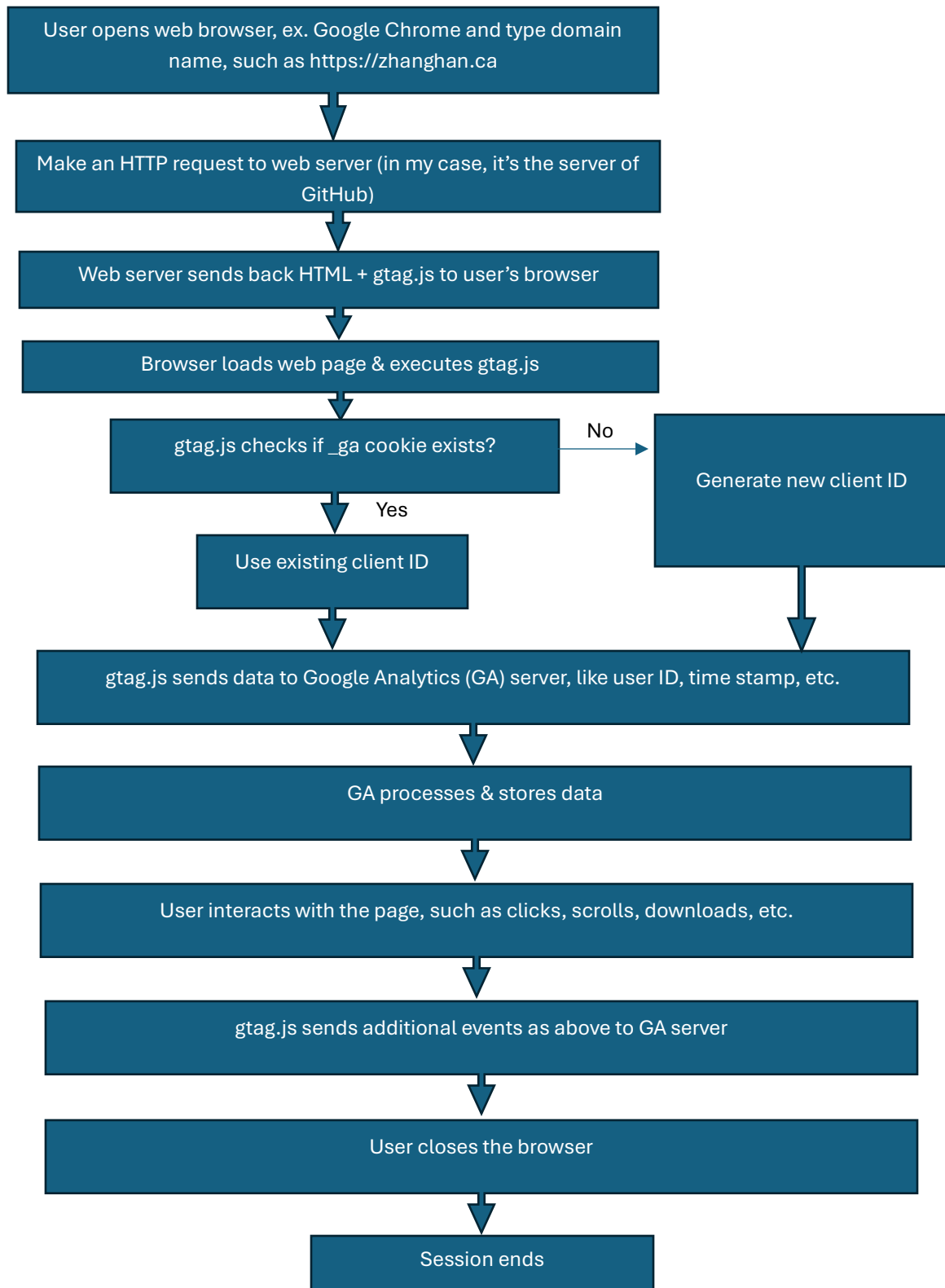
This 3-step process is generally named purchase funnel as shown below:
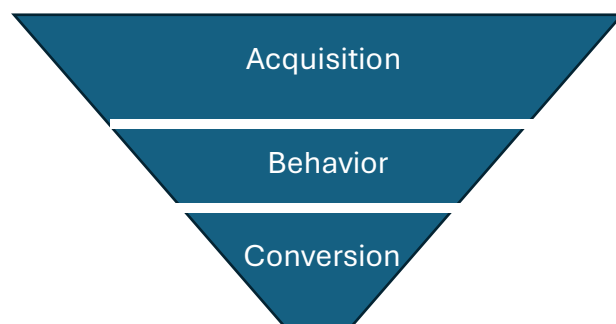
Knowing about the details of each stage will help us put our resources to the most valuable target group (like users from a certain country / province, or from a certain channel, like Facebook) to boost the KPI of an entity, such as sales for an e-commerce or enrollments for a school.

Therefore, in this project, I will deploy this funnel process in Google Analytics on my personal website (https://zhanghan.ca). Since there is neither product page nor sales page on my web, the data collected for the last step (conversion) will probably be none.
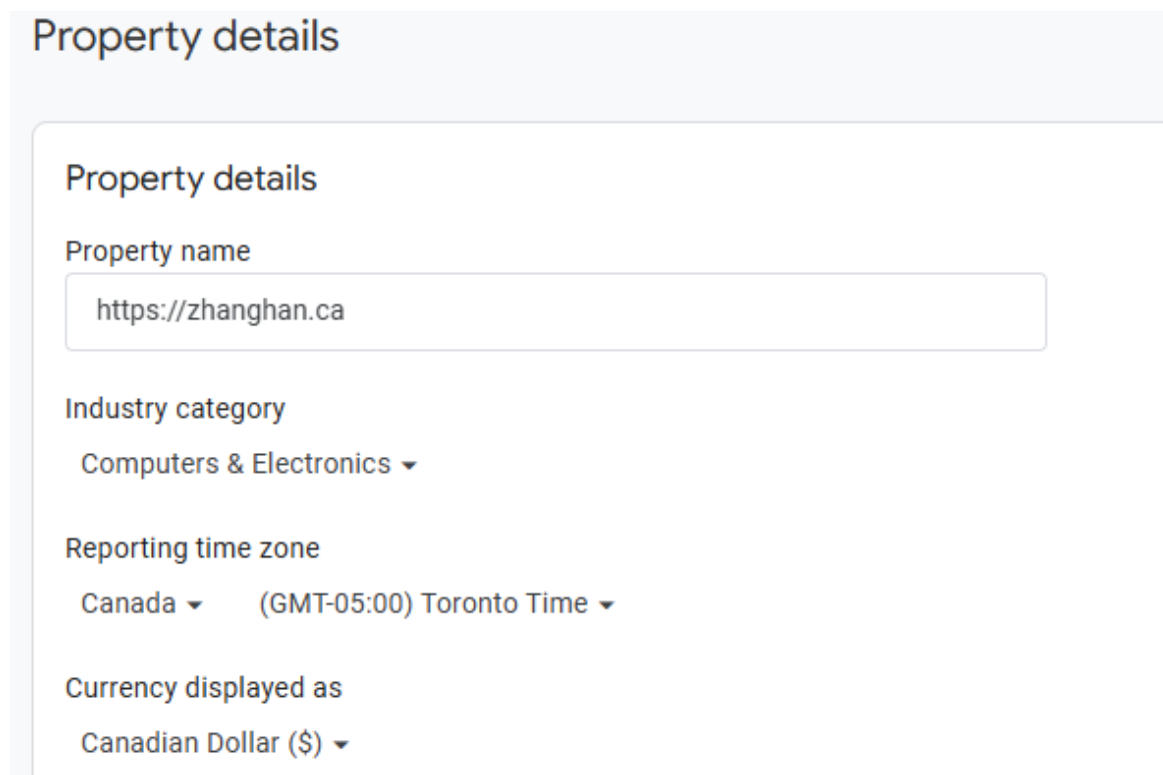
Because HTTP is stateless, which means that the server will not memorize the client's data. Therefore, we'll need a file to keep a record of these info - cookie is just this kind of file, which is a text file sent by the server to the client's browser. It will record the client's data such as user id, IP address, visiting time stamp, etc. Google Analytics is a powerful tool which uses cookies to collect users' data by planting a tag (gtag.js) in the users' computer. When a user visits the website, the gtag.js library will collect and send the data to Google server. The following diagram shows what happens in each stage from the opening of the web browser of a user to the end of that.

```mermaid
flowchart TD
    A[User opens web browser, ex. Google Chrome and type domain name, such as https://zhanghan.ca]
    B[Make an HTTP request to web server (in my case, it's the server of GitHub)]
    C[Web server sends back HTML + gtag.js to user's browser]
    D[Browser loads web page & executes gtag.js]
    E[gtag.js checks if _ga cookie exists?]
    F[Generate new client ID]
    G[Use existing client ID]
    H[gtag.js sends data to Google Analytics (GA) server, like user ID, time stamp, etc.]
    I[GA processes & stores data]
    J[User interacts with the page, such as clicks, scrolls, downloads, etc.]
    K[gtag.js sends additional events as above to GA server]
    L[User closes the browser]
    M[Session ends]

    A --> B --> C --> D --> E
    E -->|No| F
    E -->|Yes| G
    F --> H
    G --> H
    H --> I --> J --> K --> L --> M
```

Here comes my steps:

1. Build up a website

2. Create a Google Analytic account on https://analytics.google.com

3. Create a property under this account

   The property is pretty much the webs or apps that I owned, and which will be tracked and analysed in the future. I can set up at most 2000 properties. As shown in the following screenshot, here I need to give a name, select an industry, and set up the time zone and the currency of my property.

   ## Property details

   ### Property details

   Property name

   https://zhanghan.ca

   Industry category

   Computers & Electronics ▾

   Reporting time zone

   Canada ▾   (GMT-05:00) Toronto Time ▾

   Currency displayed as

   Canadian Dollar ($) ▾

4. Create a data stream under each property

   Data stream is the flow of data which comes from the web sites or apps that users visited and sent to the server of Google Analytics by the tags in the web pages. There are 3 types of data stream: Web (for websites), iOS (for iOS apps), and Android (for Android apps). In my case, I'll just have the Web stream. Each stream comes from only one property, with only one ID.

   In Goolle Analytics, select the "Admin" on the left pane (a little gear icon), click "Data collection and modification", then click "Data streams", then click "Add

stream". Since I will track my website, so I will select "Web". Enter the URL of my web site – https://zhanghan.ca. Then give a proper name, as shown below.

Set up your web stream

Website URL		Stream name
https:// ▾	zhanghan.ca	My Website

If everything works well, we'll have a page like this where I'm assigned a unique stream & measurement ID, which will be used in the tags to identify my stream.

✕ Web stream details

⊘ Data collection is active in the past 48 hours.

Stream details   ✎

STREAM NAME	STREAM URL	STREAM ID	MEASUREMENT ID
Han's Studio	https://zhanghan.ca	100⸍⸍⸍⸍⸍	G-P9⸍⸍⸍⸍⸍⸍ ⧉

Events

✦ **Enhanced measurement**   🔵✓
Automatically measure interactions and content on your sites in addition to standard page view measurement.
Data from on-page elements such as links and embedded videos may be collected with relevant events. You must ensure that no personally-identifiable information will be sent to Google. Learn more

Measuring: 👁 Page views   ✦ Scrolls   🅱 Outbound clicks   + 4 more   ⚙

5. Plant the tags (gtag.js) into the web pages
   With the data stream configured, it's time to plant the tags. In the same page as shown above (Web stream details), at the bottom, click "View tag instructions", as shown below:

Google tag

🏷	**Configure tag settings** Configure your Google tag behavior, including cross-domain linking and internal traffic. Learn more		＞
‹·›	**Manage connected site tags** Load tags for additional properties or products using this stream's on-page Google tag. Learn more	0 connected	＞
⟨⟩	**View tag instructions** Get instructions for how to implement your Google tag for this data stream. Learn more	✓ Data flowing	＞

Take a screenshot

Once click, I'll have 2 options, install the tags by CMS or manually. CMS - Content Management System is a software (or platform) helping us build websites, such as WordPress or WIX.

Since I didn't use any CMS, I'll choose the second one – install manually, as shown below. Here the JavaScript code was generated automatically by GA, the only thing I need to do is to copy & paste this code block in each of my web page under the <head> tag.

In general, this code will do the following things:

1) Load the library from https://www.googletagmanager.com/gtag/js. It contains the methods "listening to" the events happened in my web pages. If an event (such as click) is triggered, it will automatically send a record to the data layer (a kind of container). The "async" implies that the library will be loaded in an asynchronous way such that it will not affect the ordinary HTTP traffic between the client and server.

2) Define a datalayer. It is an empty JS array works as a container in the client's browser to store the pushed data temporarily, such as date or ID, before they are sent to the server of GA. The gtag() method is defined in gtag.js and will simply "push" these data into the datalayer (here it's like a function overloading). Other events such as 'page_view', 'click' will be triggered and pushed into the datalayer automatically.

3) The gtag('js', new Date()) and gtag('config', 'G-***') simply push the date and tracking ID into the datalayer. And these data will be sent along with other events, such as 'page_view', 'scroll' or 'click', etc. to the server of GA in a certain frequency.

    In the end, I will click "Test" to see if the tag in each page works properly. If all good, there will be a check mark before the URL. Till now, a major part of this project is done. That means the GA can collect the users' data from my website at this moment.

6. Configure the consent settings

    In most countries or regions, by law I must ask the consent of users to collect the data, such as North America and Europe. Consent is nothing but to notify the users that I'm going to collect the data under a series of agreement. They can choose to "Allow all", "Deny" or "Customize" (partially allow). So, I will NOT collect the user's data without the consent setting. Following is an example of a typical user interface (UI) of a consent banner.

Consent	Details	About

This website uses cookies

We use cookies to personalise content and ads, to provide social media features and to analyse our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

Deny	Customize >	Allow all

Now I'll go back to the "Admin" page of Google Analytics, under "data collection and modifications", click "Consent settings", select a data stream, then click "Set up consent mode". As shown in the following:

Since I've already set the consent mode, it shows that it's active, but I'll still "proceed anyway"

Input my URL and click "Check website"
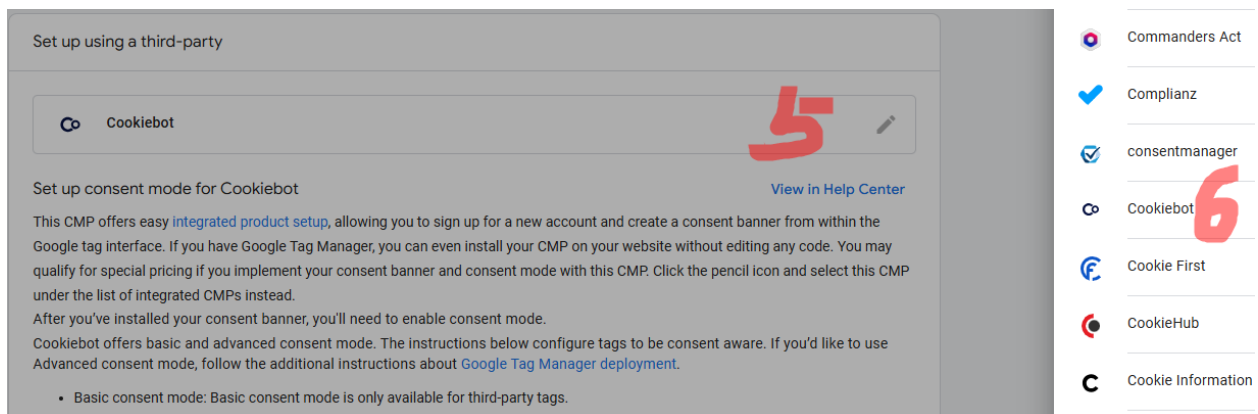
If the website is good, a new page will show up letting me choose a third-party Consent Management Platform (CMP), which is a tool helping me to configure all about the consent.

So, when I click the "pencil" icon, a list of CMP certified by Google will be shown. I chose "Cookiebot", since it has free tier. But I can set up only one domain name (website), and in each website, I can have at most 50 pages. Well, that doesn't bother me at all. So, I'll get it! First things first, I'll need to sign up to get an account from https://www.cookiebot.com.

Now I'm about to leave the Google Analytics temporarily and play with Cookiebot. Once signed in, in the "Implementation" page, there are 4 sub sections (the first 3 is important to me):

Implementation

CMP Banner   Cookie declaration   Google Consent Mode   A/B Testing

Implement CMP banner

Implement the Cookiebot CMP banner manually by adding the script tag below or follow one of our **installation guides.**

WordPress ↗   **Google Tag Manager** ↗

- CMP Banner supplies a code block (script) showing a banner as shown in step 6 above - Configure the consent settings. It will show a banner with user interactions such as "Allow all", "Deny", "Customize". I'll copy & paste this script under <head> tag, it must be BEFORE the gtag.js.

  `<script id="Cookiebot" src="https://consent.cookiebot.com/uc.js" data-cbid="64f3e6e5-b9ff-4796-********"  type="text/javascript"></script>`

- Cookie declaration will show the details of cookies, e.g. it shows a list of cookies being used on your website along with their categories (e.g., necessary, preferences, statistics, marketing) and purpose (e.g., to track user behavior, preferences, etc.).

- Google Consent Mode is an API which Cookiebot CMP integrates with. When you have installed the Cookiebot CMP tag directly on your site, you can set default (prior consent) settings via the Google consent API-integration by adding the following code snippet to your website template.

Script tag:

```
<script data-cookieconsent="ignore">
    window.dataLayer = window.dataLayer ||
[];
    function gtag() {
        dataLayer.push(arguments);
    }
    gtag("consent", "default", {
        ad_personalization: "denied",
        ad_storage: "denied",
        ad_user_data: "denied",
        analytics_storage: "denied",
        functionality_storage: "denied",
        personalization_storage: "denied",
        security_storage: "granted",
        wait_for_update: 500,
    });
    gtag("set", "ads_data_redaction", true);
    gtag("set", "url_passthrough", false);
</script>
```

Now let me return to the Consent settings of Google Analytics, if all set properly, it will show "Receiving consent signals" and "Active".

Select a data stream:

Han's Studio ▾   ✅ **Receiving consent signals**

Behavioral analytics consent signals   ⊘ Active ⌃

This signal is used for audience measurement (to measure users and engagement with websites or apps) and behavioral modeling (to fill in the measurement gaps for users who decline consent). Learn more about behavioral analytics consent signals.

CONSENT SIGNAL STATUS

⊘ **Analytics cookie consent signals active**
   The analytics_storage parameter indicates if cookie data can be collected for behavioral analytics purposes (other modeling prerequisites apply).

IMPACT
User & event reporting   Behavioral modeling

Now it's time to verify if the consent settings work or not! Let me open google chrome and enter https://zhanghan.ca… So nervous!

This is what it shows, perfect! It worked as I expected!

Cookiebot
by Usercentrics

Consent	Details	About

This website uses cookies

We use cookies to personalise content and ads, to provide social media features and to analyse our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

Necessary	Preferences	Statistics	Marketing

Deny	Allow selection	Allow all

La ville de Sherbrooke – photo by me

Developed by: Han Zhang

Phew, till now the setting part is done (I'm very tired), I'll wait for a couple of days to accumulate some data.

10 days later ....

I played with my web site a little bit, here comes the report ((I'm so exciting):

After about 10 days, there are already some data & events recorded (not just visited by me!) I'll show some charts generated in Google Analytics.

To prove that the deployment aligns with the funnel model in the very beginning, I'll just follow the steps of Acquisition – Engagement – Conversion

From 2025-01-05 to 2025-02-02, the data is:

- Acquisition (who are the users):

This time-serious line chart (one of the charts) shows that the number of active users is 16, among which 13 are new users. There was a peak (about 11 users) on 26 Jan.

Acquisition overview

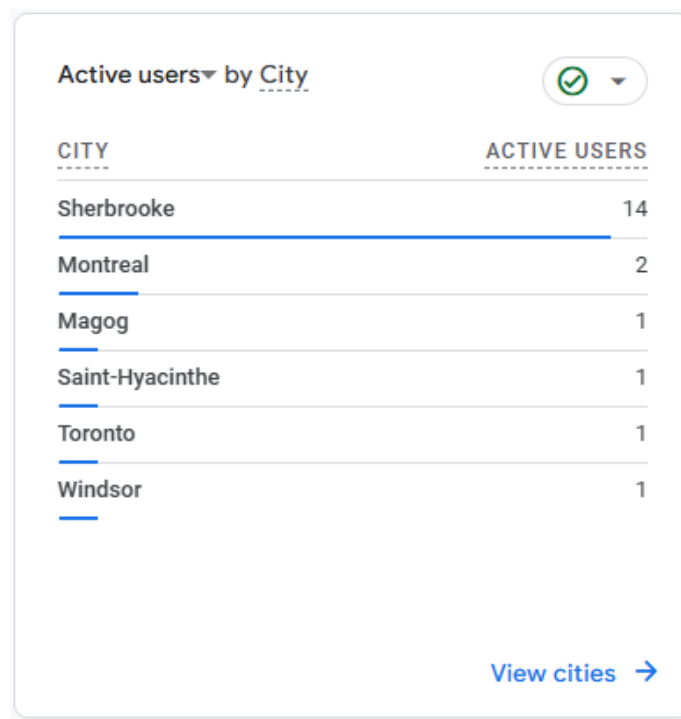Active users ⑦   New users ⑦

16   13
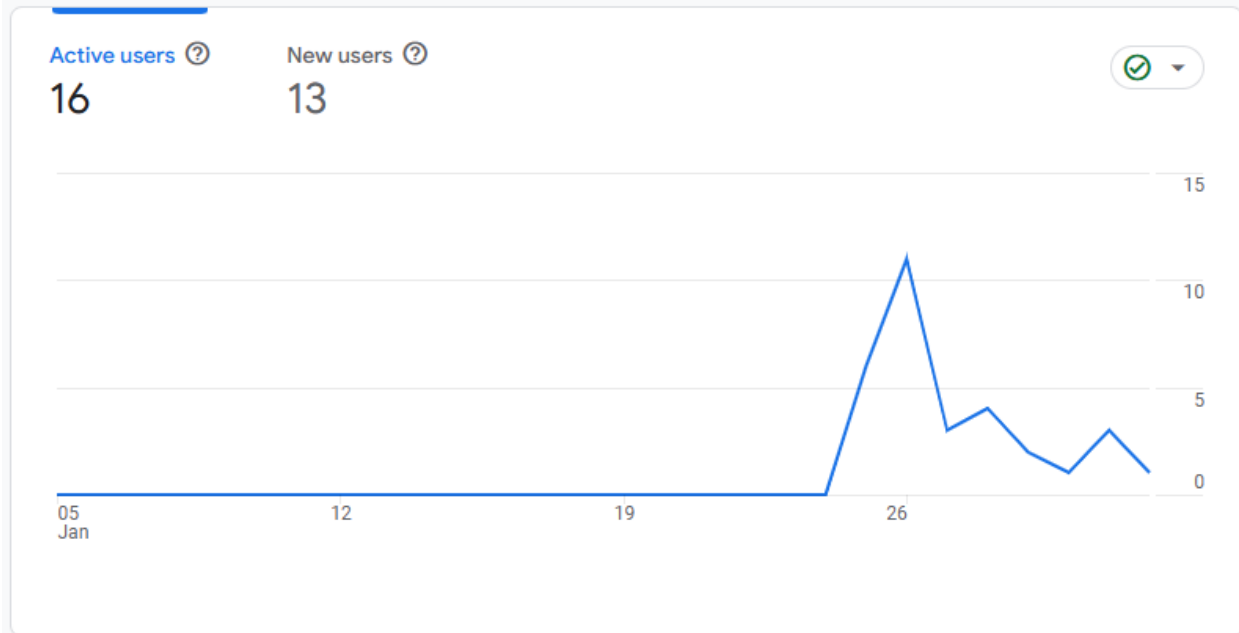
✓ Where are they from?

As shown in this table below, among the 16 users, 15 are from Canada, 1 is unknown. – Till today, I have no user from outside of Canada (if the 1 unknown is also from Canada).

Active users▾ by Country

COUNTRY	ACTIVE USERS
Canada	15

View countries →
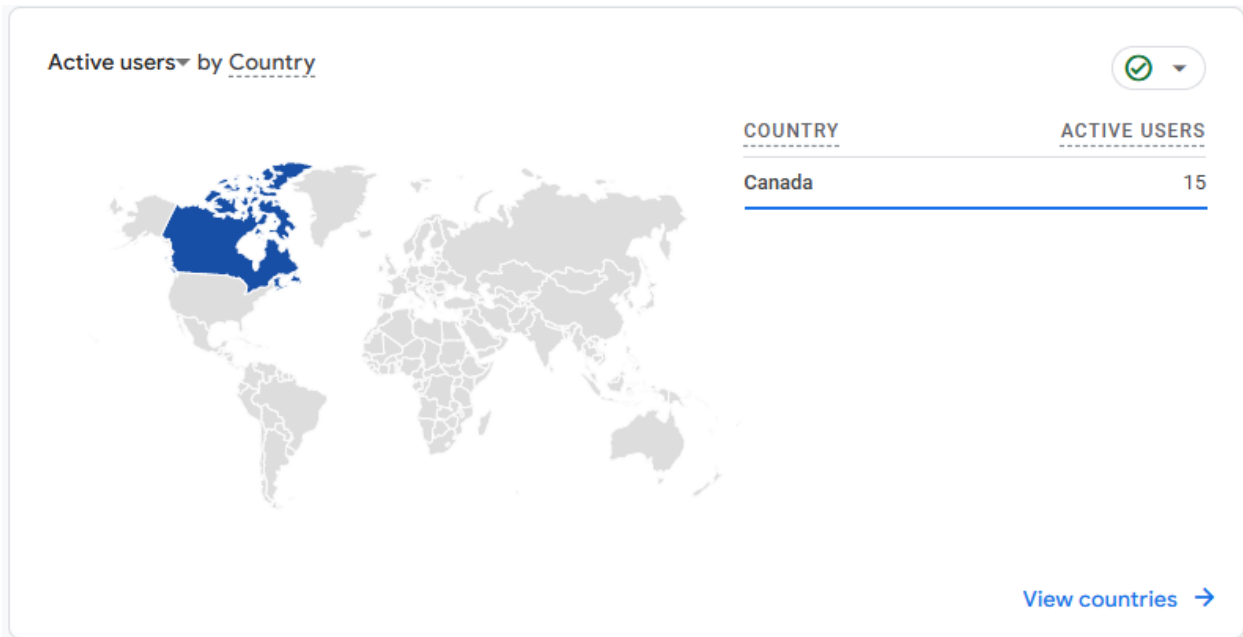
But where in Canada? The following bar chart shows the geographical distribution of the users. The top one is Sherbrooke – with 14 users, the last one is Windsor with 1 user. The sum of count in each one is bigger than the total -16, this is no strange, since I visited my web page using different IP addresses several times (at least an IP from Montreal, and 2 IPs from Sherbrooke are all generated by myself).

Active users▾ by City

CITY	ACTIVE USERS
Sherbrooke	14
Montreal	2
Magog	1
Saint-Hyacinthe	1
Toronto	1
Windsor	1

View cities →

✓ What languages do they speak?

The following bar chart shows that there are 11 English users, 5 French users. This figure is generated based on the language setting of users' browser (or PC / Tablet). If they are francophone but the language setting is in English, they are still categorized into English.
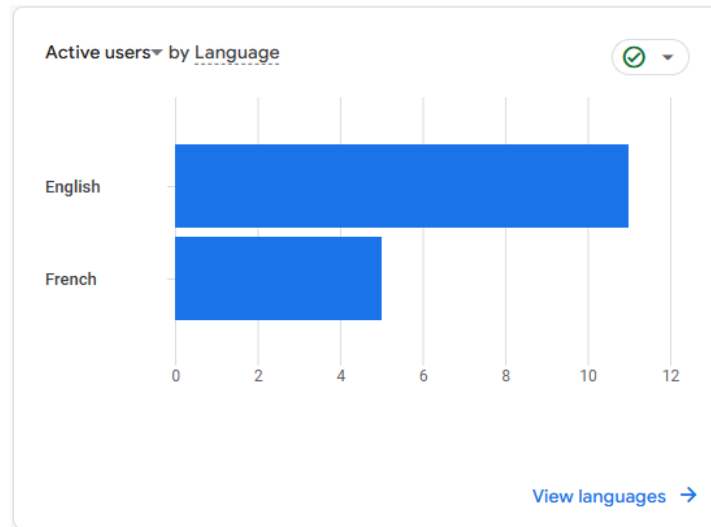

✓ Which channel do they come from?

As shown bellow, 13 users came directly by entering URL into the web browser; 3 came from LinkedIn; 1 from Facebook; 1 from Google tag assistant referral. Be careful, the sum of each category might not be equal to the total, since a user come from a channel (say LinkedIn) and then input the URL into the web browser and vice versa might be treated as 1 user, but 2 channels (or 2 sessions) - yes, I was that boring user! And GA might attribute the channel to Direct instead of the original source (that's why 1 user from LinkedIn was categorized as Direct). In one word, there is overlap. The avg. engagement time was 4m 58s, where the longest one is 13m 38s.

	First user prim...Channel Group) ▾	Session source ▾ ✕	↓ Total users	New users	Returning users	Average engagement time per active user	Engaged sessions per active user	Even All ev
	Total		16 100% of total	13 100% of total	11 100% of total	4m 58s Avg 0%	2.44 Avg 0%	10
1	Direct	(direct)	13	10	8	4m 20s	2.00	
2	Direct	linkedin.com	2	0	2	3m 27s	3.00	
3	Direct	tagassistant.google.com	1	0	1	13m 38s	2.00	
4	Organic Social	linkedin.com	1	1	1	40s	2.00	
5	Organic Social	m.facebook.com	1	1	1	1m 05s	2.00	
6	Referral	tagassistant.google.com	1	1	0	55s	1.00	

✓ What technology do they use?

As shown below, all 16 users are from the web, within whom 62.5% are from desktop, 37.5% are from mobile (tablet / smartphone, etc.). The most popular web browser they used is Chrome. The most popular Operating System is Windows – knowing about this might not directly boost the KPI, but that can help us optimise the performance and functionalities of our website to fit for users' devices.
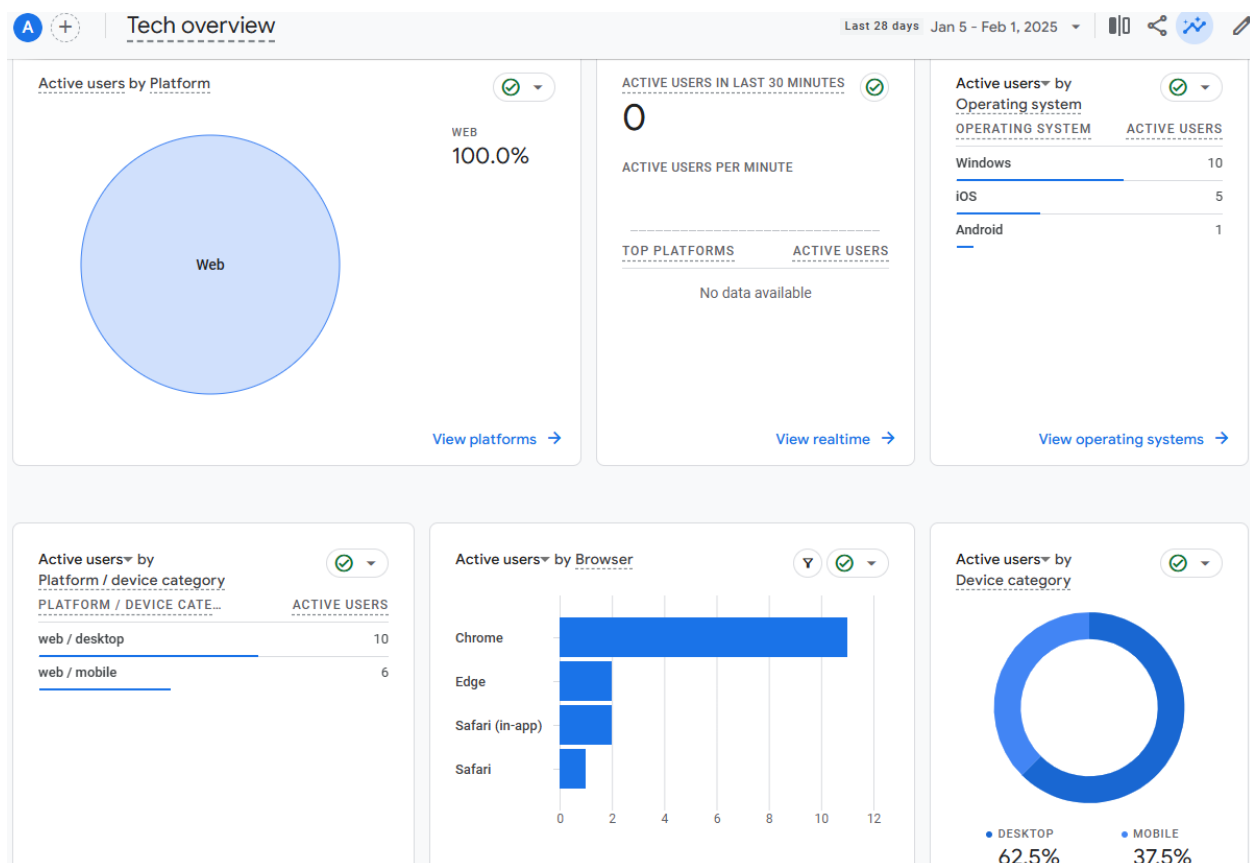
- Behaviour (Engagement):

In the second phase of the funnel, the engagement data will explain what the user did in my website. The following line chart has a general overview showing how the engagement time distributed between Jan. 05 till today (Feb. 02), where the avg. engagement time is 4m 58s, and the ag. Engagement session per active user is 2.4. Jan 26 has max. average engagement time (around 8.5m).

Engagement overview
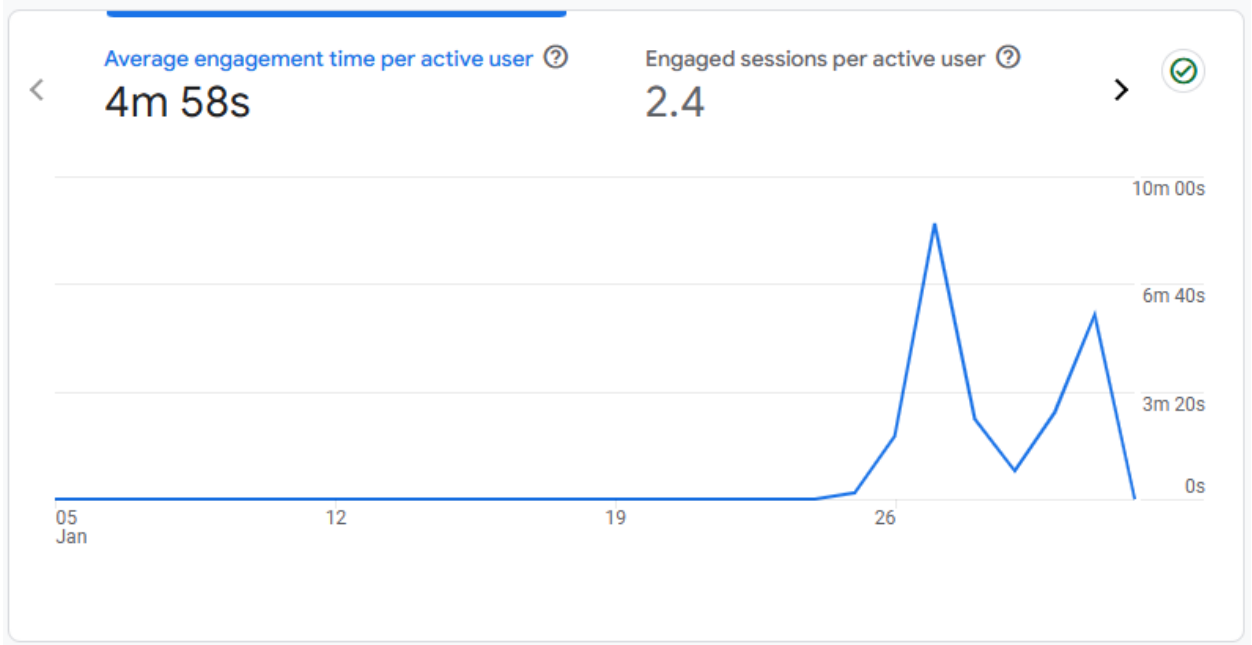
The 16 users have made 280 views and 862 events during this period (I can toggle between the tab "Views" and "Event count").

Views ?	Event count ?
280	862
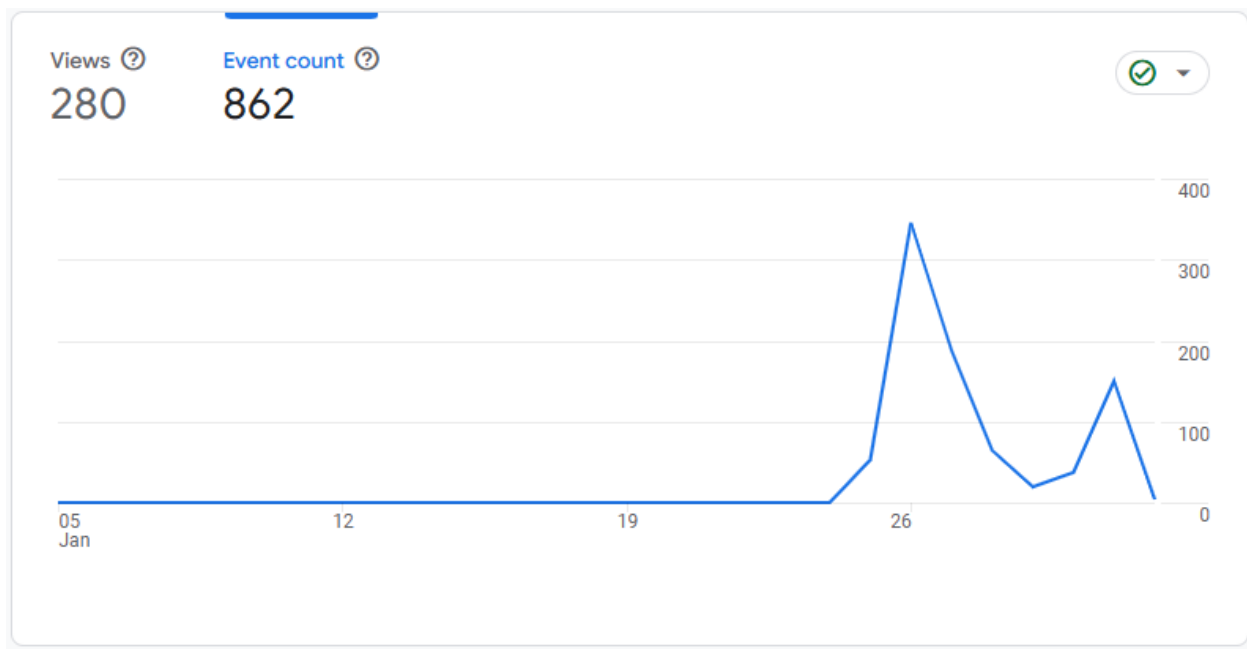
✓ What did they do?

The following table illustrates the detail of the 862 events, where the top one is the "page_view" event, with 280 counts by 15 users; the last one is the "file_download" event, with 1 count by 1 user. This is reasonable, since in my website, there are not too many elements, most of them are just texts, therefore, the things user can do mostly is to view the page. And there is only one button element to download my cv (that's why it's the least count event) – or maybe this button is not too attractive to the users.

	Event name	Event count ↓	Total users	Event count per active user	Total revenue	
	Total	862 100% of total	16 100% of total	53.88 Avg 0%	$0.00	
1	page_view	280	15	18.67	$0.00	⋮
2	user_engagement	279	14	19.93	$0.00	⋮
3	scroll	227	15	15.13	$0.00	⋮
4	session_start	49	16	3.06	$0.00	⋮
5	first_visit	13	13	1.00	$0.00	⋮
6	click	8	5	1.60	$0.00	⋮
7	form_start	5	4	1.25	$0.00	⋮
8	file_download	1	1	1.00	$0.00	⋮

Plot rows   Search…   Rows per page: 10 ▼   1-8 of 8

✓ Which pages are attractive?
As shown below, the /myjob.html has most views – 91 by 9 active users. The last one is /project_3.html page

| Page path and screen class ▼	Event name ▼ ✕	↓ Views	Active users	Views per active user	Average engagement time per active user	Event count	Key events
Total		280 100% of total	16 100% of total	17.50 Avg 0%	4m 58s Avg 0%	862 100% of total	13.00 100% of total
1 /myJob.html	page_view	91	9	10.11	0s	91	0.00
2 /index.html	page_view	50	11	4.55	0s	50	0.00
3 /	page_view	48	12	4.00	0s	48	0.00
4 /myProject.html	page_view	38	8	4.75	0s	38	0.00
5 /aboutMe.html	page_view	21	6	3.50	0s	21	0.00
6 /project_4.html	page_view	9	4	2.25	0s	9	0.00
7 /project_1.html	page_view	7	3	2.33	0s	7	0.00
8 /contactMe.html	page_view	6	4	1.50	0s	6	0.00
9 /project_2.html	page_view	5	2	2.50	0s	5	0.00
10 /project_3.html	page_view	3	2	1.50	0s	3	0.00

• Conversion (Monetization)
Since I have neither product page nor sales page, I have no available data in this part.

Purchase revenue by Transaction ID over time — Day

No data available
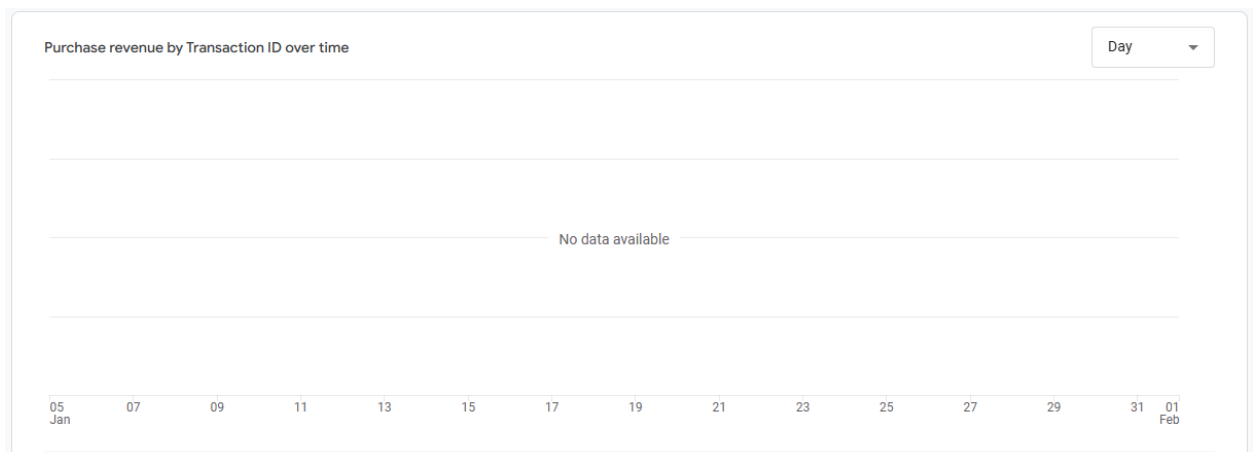
But, as a compromise, I can use the retention data instead to at least know who and how many are interested in me (since they returned), as shown below among the 13 new users, 11 returned, the retention rate on 28th Jan. was 100%.

Retention overview

New users ⑦ **Returning users** ⑦
13   11

User retention by cohort
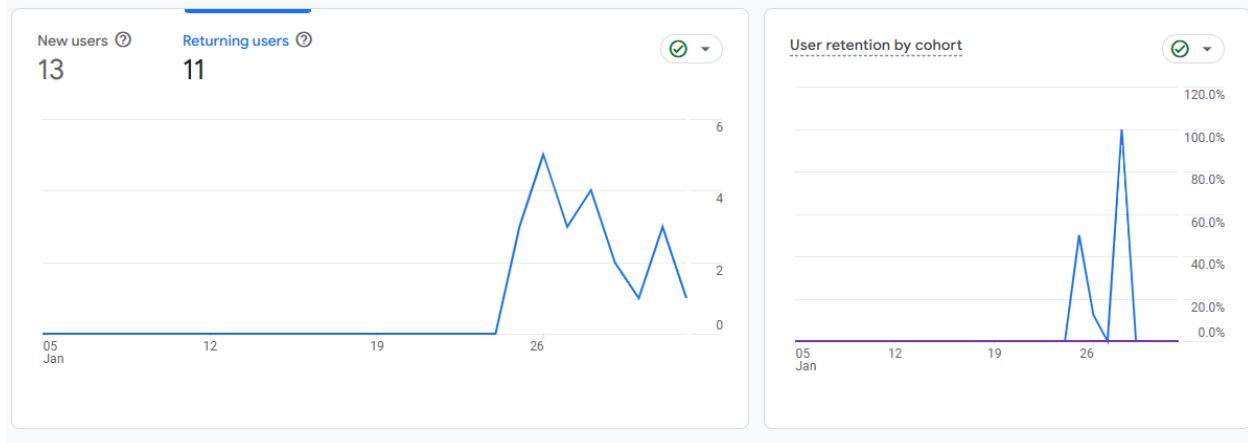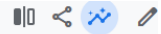
Till now, the web traffic analysis is completed (at least based on the current accumulated data). In my opinion, this can also apply to a schools' admission processes. Even though most schools have their own funnel models for the analytics of application – admission – registration cycle, all of that is based on the data that we've acquired. However, what happens before that is still a black hole. To increase the enrollments, this pre-application stats is still relevant and important.