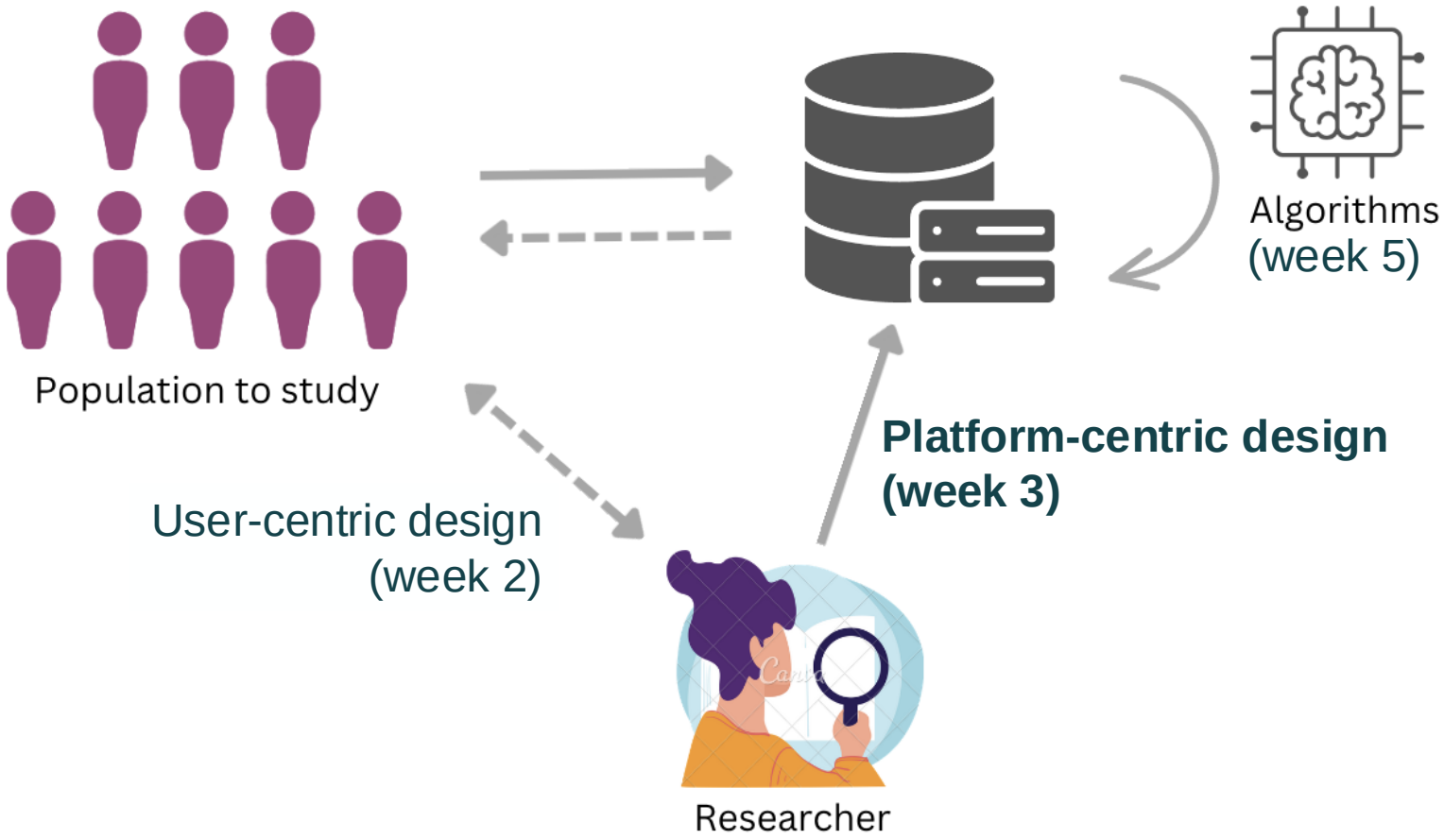
Introduction to digital trace data: Quality, ethics, and analysis

Lecture 3: Collecting platform-centric data

Javier Garcia-Bernardo
Assistant Professor
Department of Methodology and Statistics

Where are we?



Week 4: Errors in DTD
Week 6: Ethics and Legislation
Week 7: Designed big data
Week 8: Beyond DTD and Q&A

Platform-centric design

Collaboration with the organization holding the data

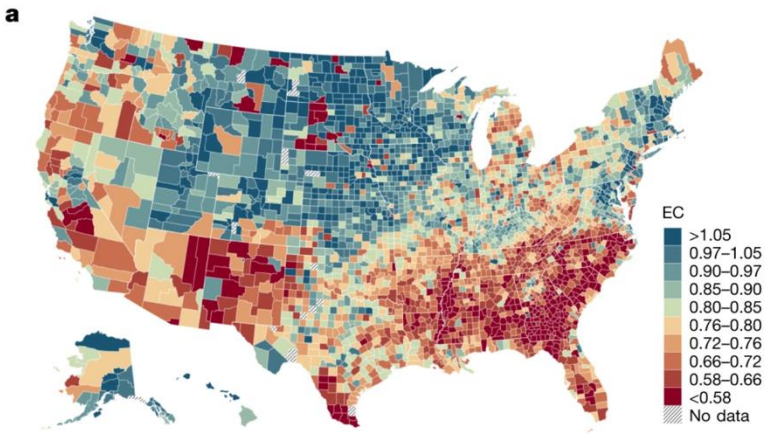
- The organization provides the data privately
- Public and private data

APIs (Application Programming Interfaces)

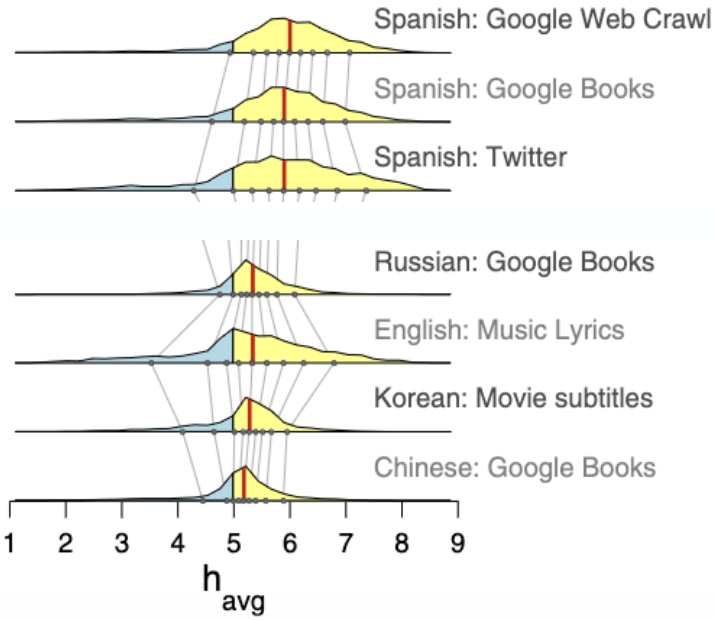
- The organization provide direct access to (parts) of the data.
- Public data only

Web-Scraping

- Download the information as shown in the website
- Public data only (at least legally)



Chetty et al, 2022, Nature



Dodds et al., 2015, PNAS

TODAY

Lecture

Explain what APIs and web scraping are (in your own words).

Understand how HTML code is structured

Distinguish between the role of robots.txt, Terms of Service and GDPR.

Understand the main advantages, challenges and legal considerations of APIs and Web Scraping.

Lab

Learn how to read robots.txt

Use APIs to extract data:

- Wikimedia
- TheGuardian

Web scraping data from NU.nl

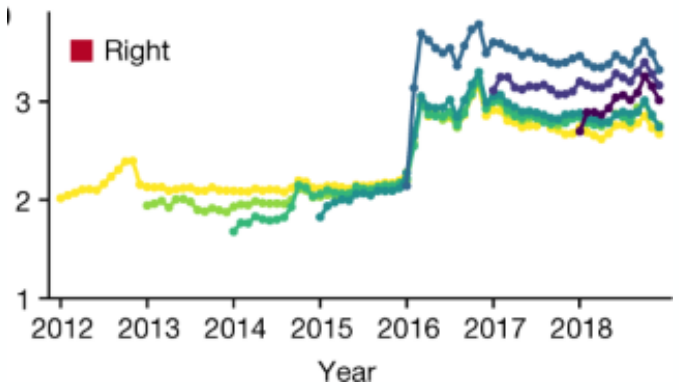
Saving text data to your hard drive

Why take a platform-centric approach?

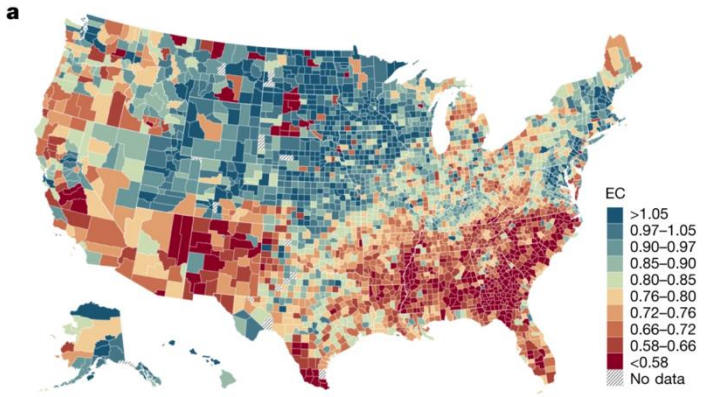
Studying aggregated effects

- Platform-specific effects

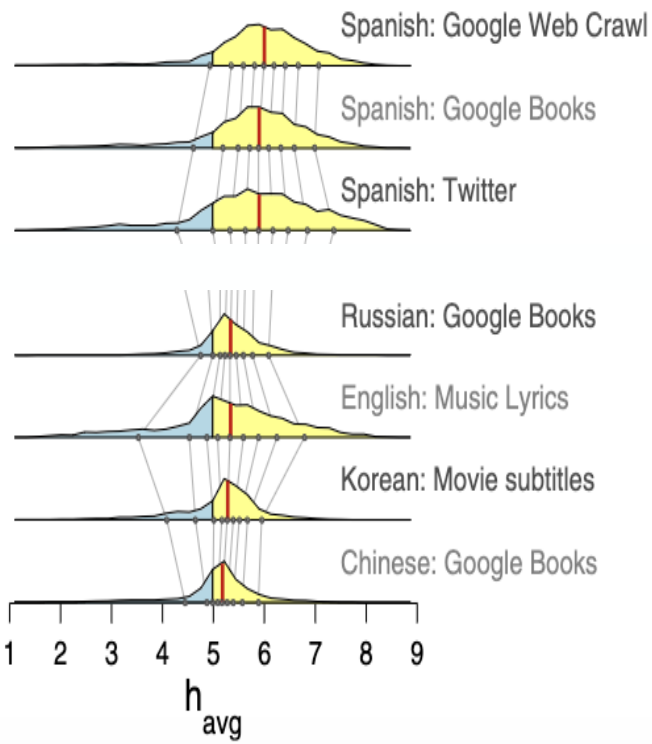
- How does X affects Y in general?
  - Big assumption here! Either that the data is representative of the target population or the studied phenomenom applies to the target population.



Waller and Anderson, 2021, Nature



Chetty et al, 2022, Nature



Dodds et al., 2015, PNAS

Application Programming Interfaces

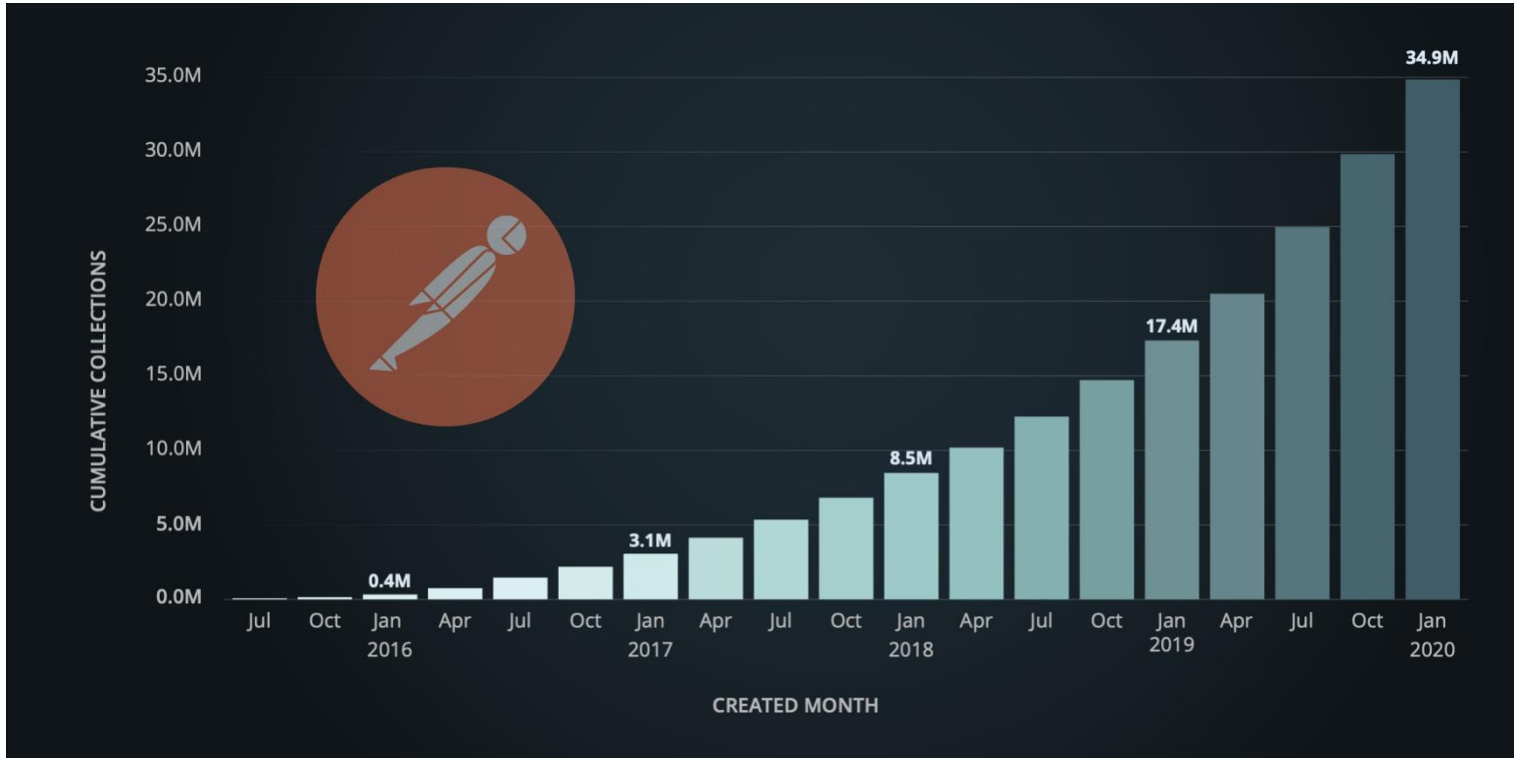
APIs

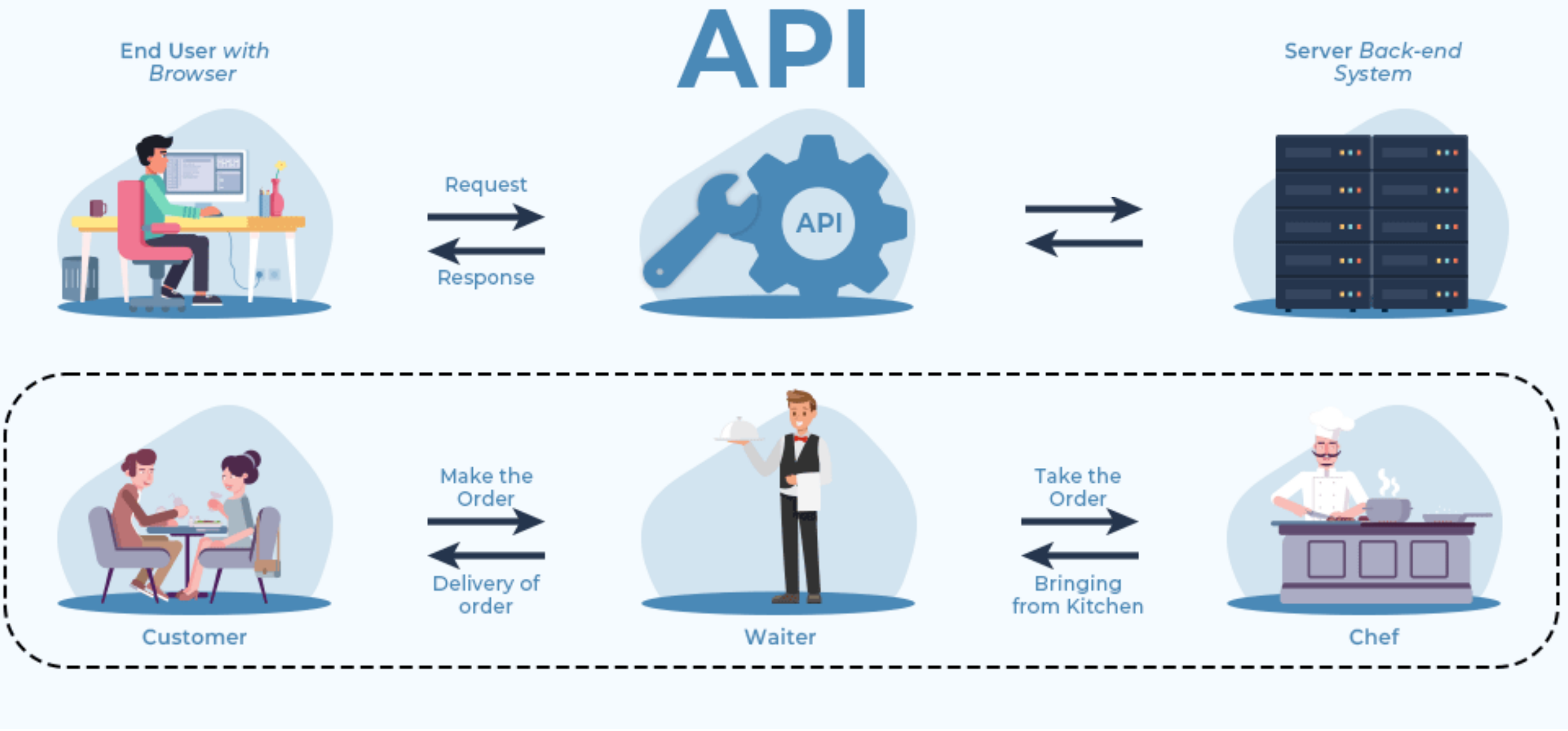
What are Application Programming Interfaces (APIs)?

APIs allow one computer (the client) to request data or services from another computer (the server). They act as a bridge between different systems. APIs provide:

1. Communication rules: APIs set rules for how computers communicate.
2. Security: APIs control who can access what information, so different users might see different data.

APIs are mainly designed for developers to use, not for researchers.





You just need to know how to ask properly.

https://www.geeksforgeeks.org/what-is-an-api/

Example use case

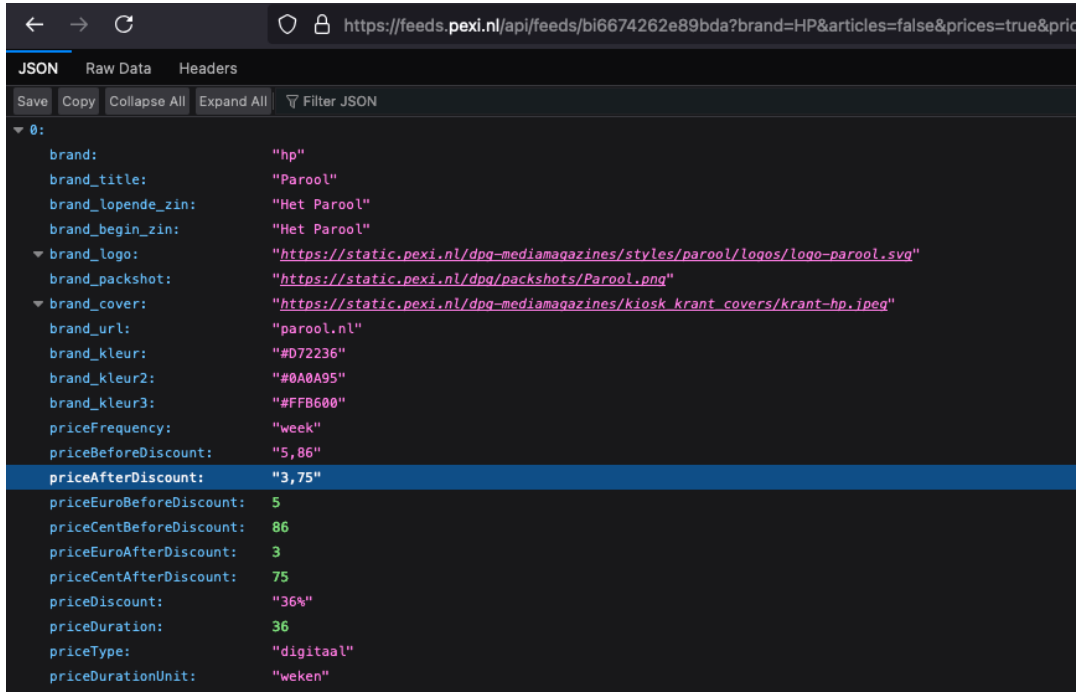
Het Parool wants to encourage readers to become members. To do so, it shows a banner at the bottom. However:

- The membership price must be accurate and up-to-date
- An independent company should manage memberships and payments.

Solution: Use an online service that provides an API to handle membership and payment processing.

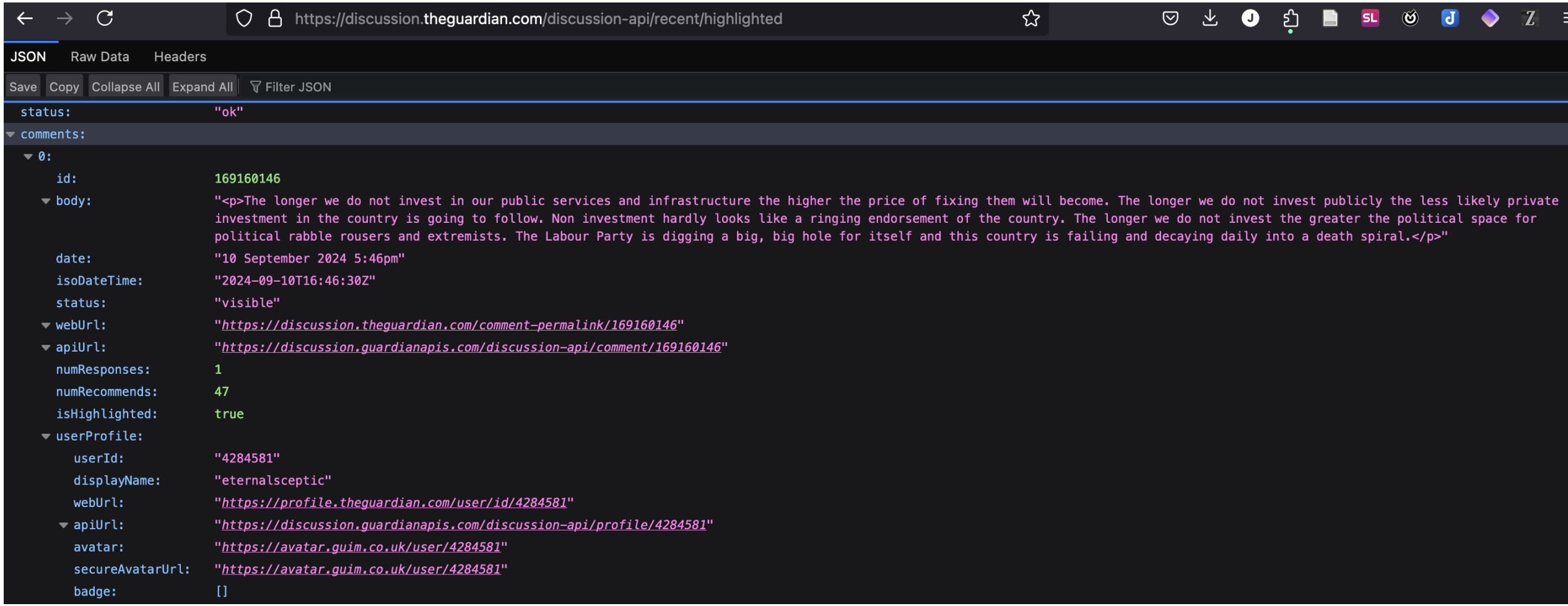
When you load the site, your browser calls the API:
https://feeds.pexi.nl/api/feeds/bi6674262e89bda?brand=HP&articles=false&prices=false&pricetype=digitaal

The API returns the data (JSON)
Your browser sets up the correct price



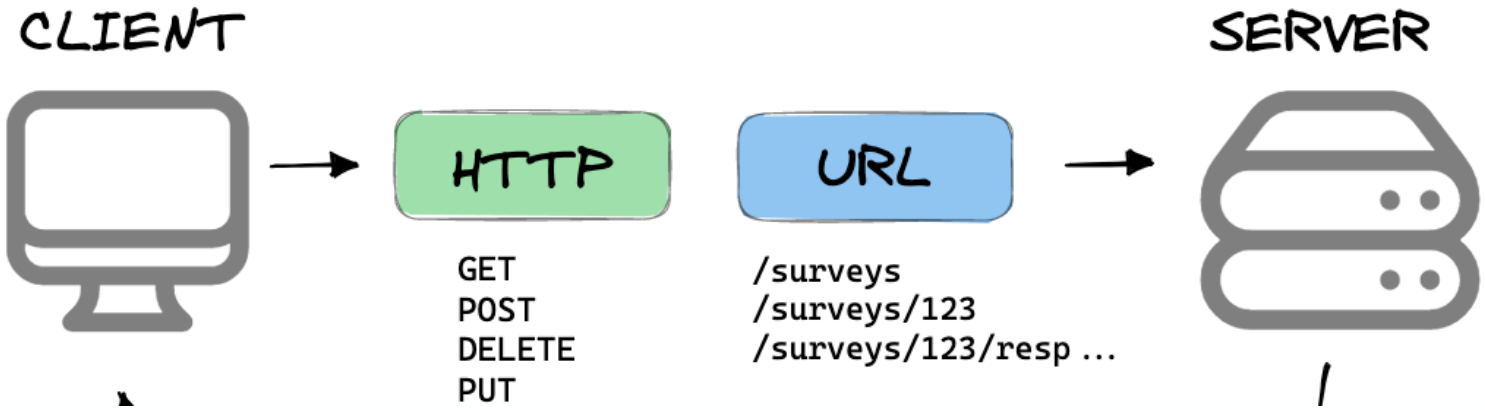
Example use case in social science

For example, to extract the comments from TheGuardian using their Discussion API



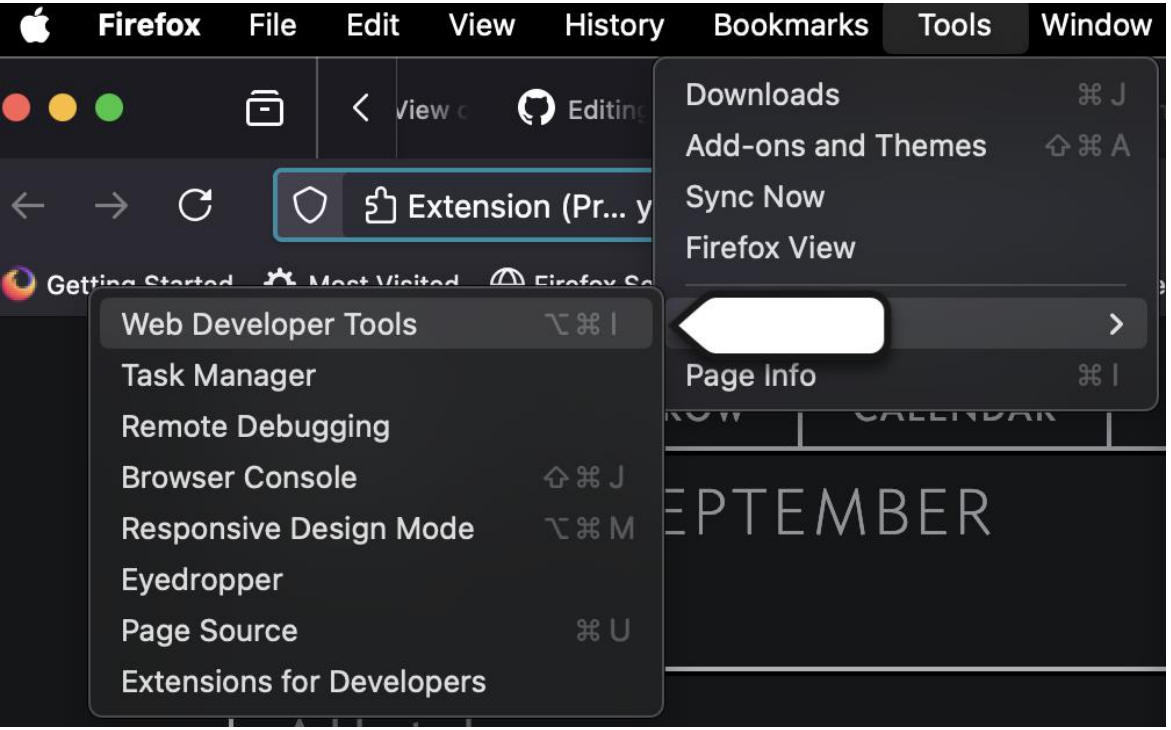
Web browsing works through HTTP(S) calls

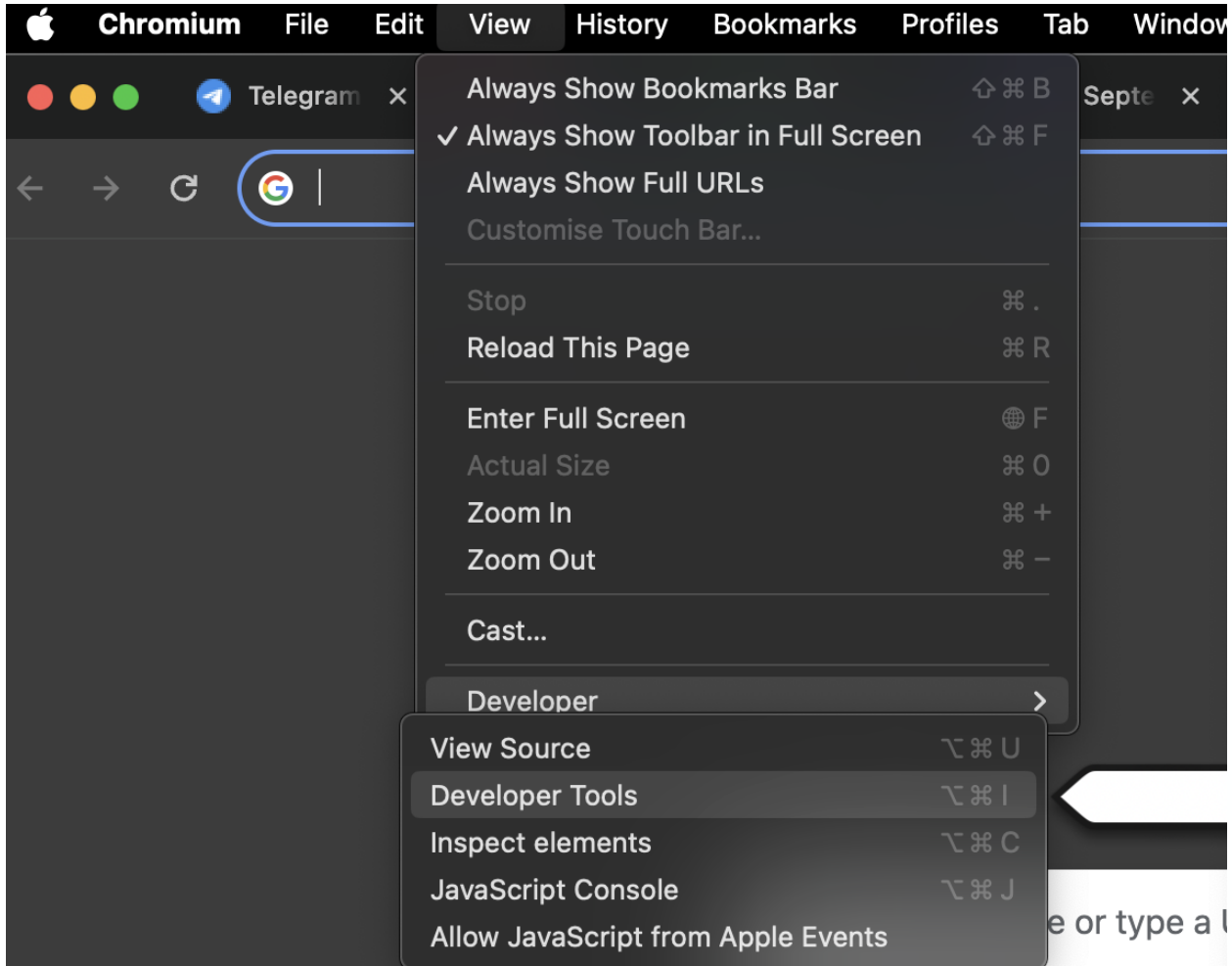
- **HTTP(S)** (Hypertext Transfer Protocol) calls = request a resource (e.g. a website, an image) from a server
  - HTTP(S) calls have *headers* and *body*, where extra information is sent to the server
- URL = location of the resource



Developer Tools in the browser

Inspect and analyze the HTML, CSS, and JavaScript of a webpage to understand its structure.
Monitor network activity to see how data is requested and received.

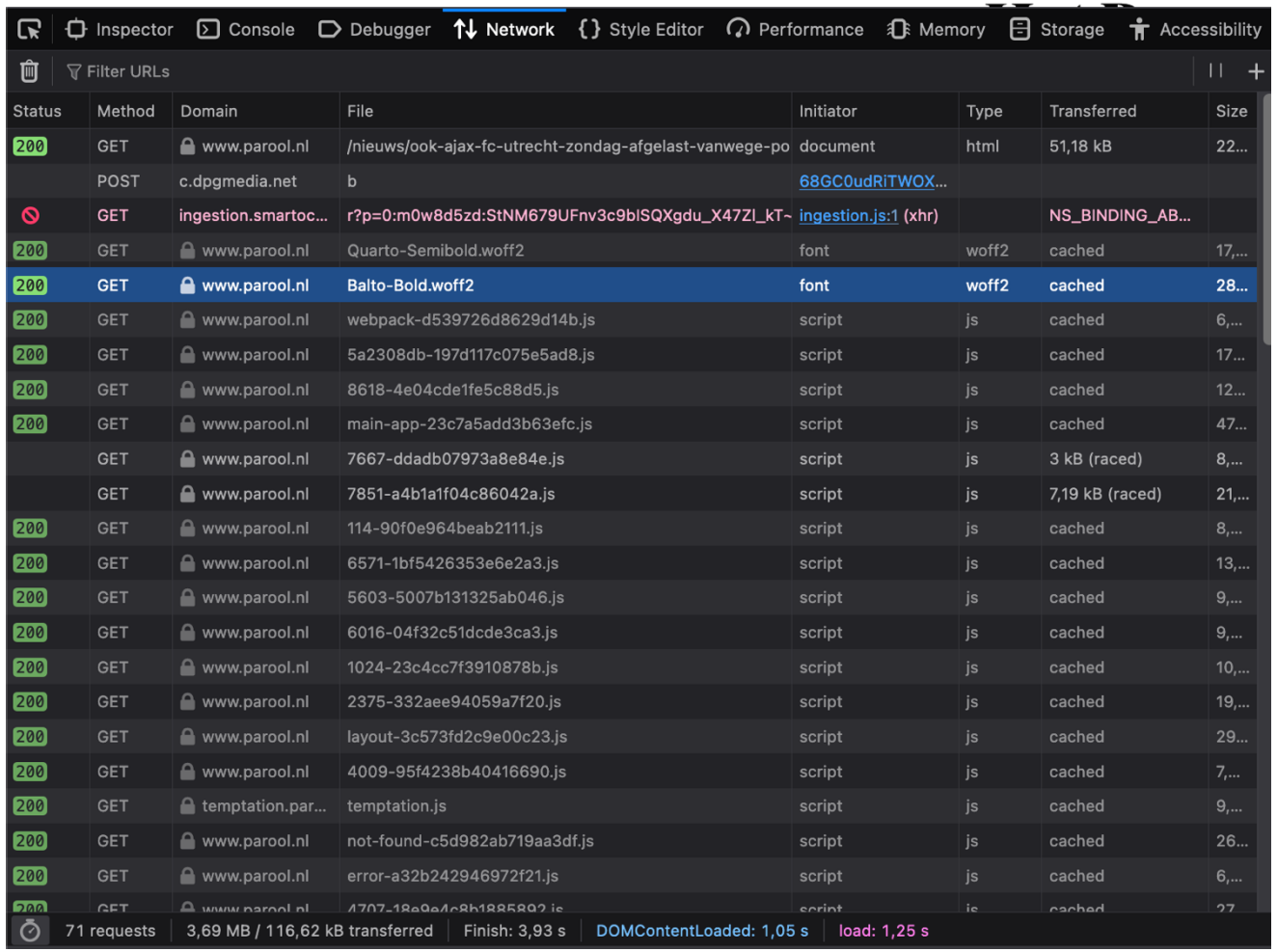


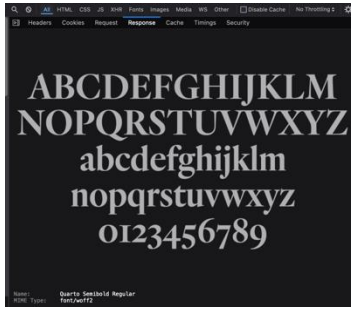


Nieuws

Ook Ajax-FC Utrecht zondag afgelast vanwege politiestaking: ‘Hiermee ontneem je supporters de kans om te laten zien dat het kan’

De Eredivisiewedstrijd tussen Ajax en FC Utrecht, die aankomende zondag in de Johan Cruijff Arena zou plaatsvinden, gaat niet door. Het is de tweede keer dat een

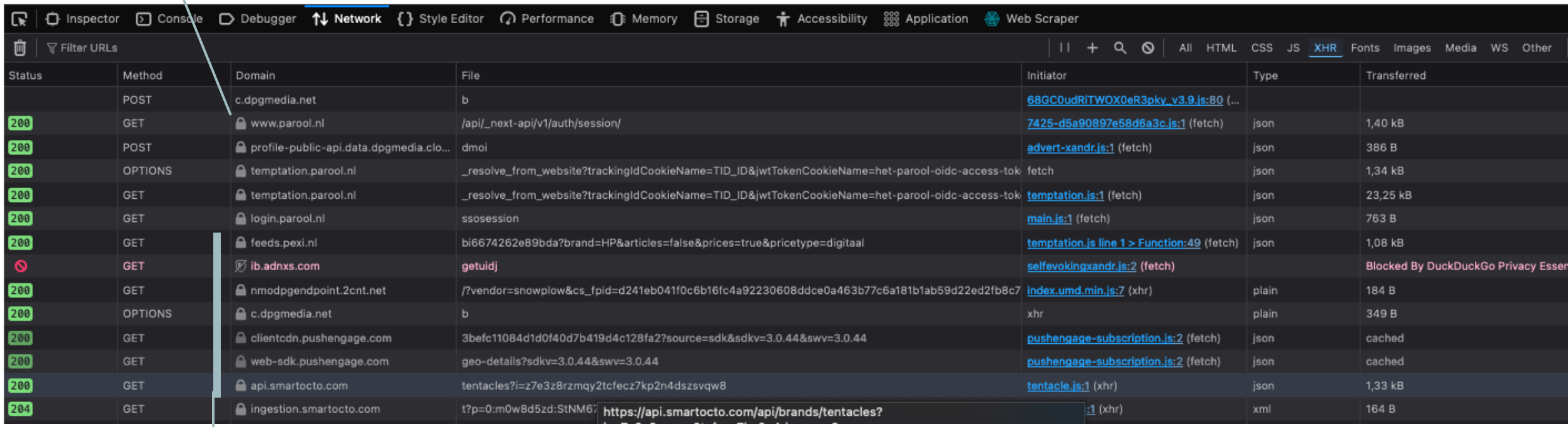




https://www.parool.nl/nieuws/ook-ajax-fc-utrecht-zondag-afgelast-vanwege-politiestaking-hiermee-ontneem-je-supporters-de-kans-om-te-laten-zien-dat-het-kan~b849073c/

Data from their servers

Ook Ajax-FC Utrecht zondag afgelast vanwege politiestaking: ‘Hiermee ontneem je supporters de kans om te laten zien dat het kan’



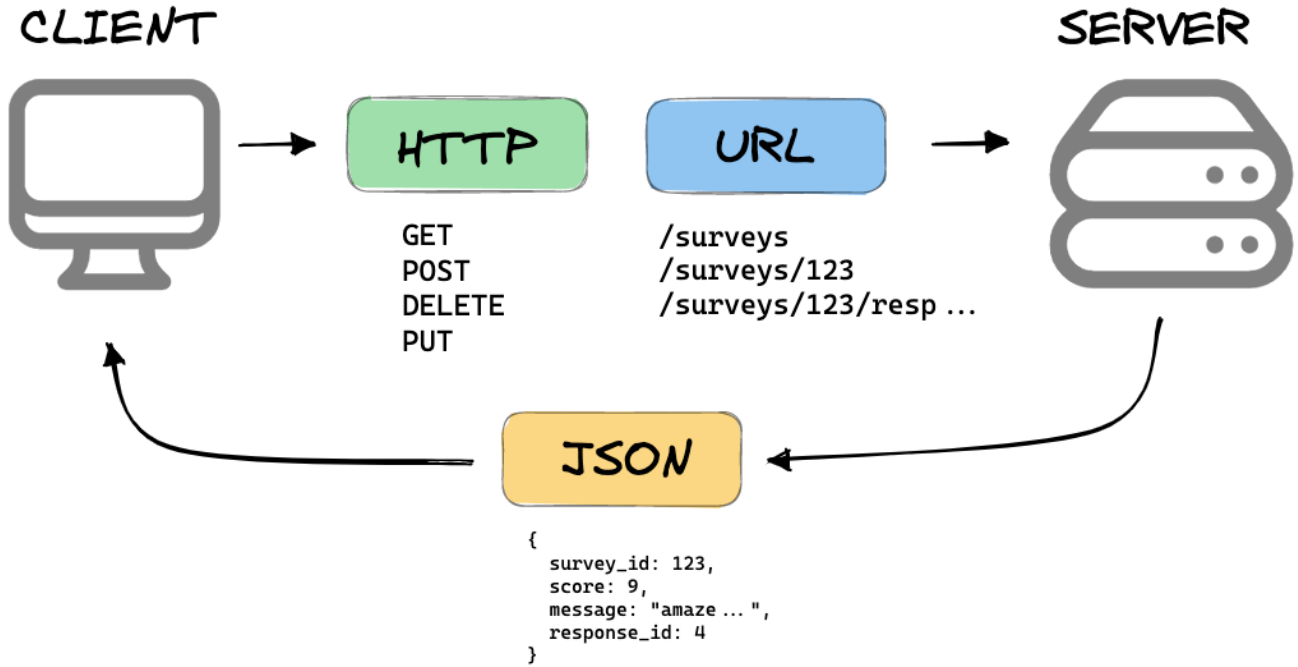
Status	Method	Domain	File	Initiator	Type	Transferred
	POST	c.dpgmedia.net	b	68GC0udRiTWOX0eR3pky_v3.9.js:80 (...		
200	GET	www.parool.nl	/api/_next-api/v1/auth/session/	7425-d5a90897e58d6a3c.js:1 (fetch)	json	1,40 kB
200	POST	profile-public-api.data.dpgmedia.clo...	dmoi	advert-xandr.js:1 (fetch)	json	386 B
200	OPTIONS	temptation.parool.nl	_resolve_from_website?trackingIdCookieName=TID_ID&jwtTokenCookieName=het-parool-oidc-access-tok	fetch	json	1,34 kB
200	GET	temptation.parool.nl	_resolve_from_website?trackingIdCookieName=TID_ID&jwtTokenCookieName=het-parool-oidc-access-tok	temptation.js:1 (fetch)	json	23,25 kB
200	GET	login.parool.nl	ssosession	main.js:1 (fetch)	json	763 B
200	GET	feeds.pexi.nl	bi6674262e89bda?brand=HP&articles=false&prices=true&pricetype=digitaal	temptation.js line 1 > Function:49 (fetch)	json	1,08 kB
	GET	ib.adnxs.com	getuidj	selfevokingxandr.js:2 (fetch)		Blocked By DuckDuckGo Privacy Esser
200	GET	nmodpgendpoint.2cnt.net	/?vendor=snowplow&cs_fpid=d241eb041f0c6b16fc4a92230608ddce0a463b77c6a181b1ab59d22ed2fb8c7	index.umd.min.js:7 (xhr)	plain	184 B
200	OPTIONS	c.dpgmedia.net	b	xhr	plain	349 B
200	GET	clientcdn.pushengage.com	3befc11084d1d0f40d7b419d4c128fa2?source=sdk&sdkv=3.0.44&swv=3.0.44	pushengage-subscription.js:2 (fetch)	json	cached
200	GET	web-sdk.pushengage.com	geo-details?sdkv=3.0.44&swv=3.0.44	pushengage-subscription.js:2 (fetch)	json	cached
200	GET	api.smartocto.com	tentacles?i=z7e3z8rzmqy2tcfecz7kp2n4dszsvqw8	tentacle.js:1 (xhr)	json	1,33 kB
204	GET	ingestion.smartocto.com	t?p=0:m0w8d5zd:StNM67	:1 (xhr)	xml	164 B

Ads, tracking, analytics, experiments (A/B testing), other services

https://www.parool.nl/nieuws/ook-ajax-fc-utrecht-zondag-afgelast-vanwege-politiestaking-hiermee-ontneem-je-supporters-de-kans-om-te-laten-zien-dat-het-kan~b849073c/

Using RESTful APIs

- RESTful APIs work using **HTTP calls** to **URLs**
  - **HTTP** (Hypertext Transfer Protocol) calls
  - URL = location of the API resource
- Returns data (typically JSON)



Examples of RESTful APIs

https://api.wikimedia.org/feed/v1/wikipedia/en/featured/2024/09/20

https://api.crossref.org/works?query.author="Javier Garcia-Bernardo"&filter=from-pub-date:2024-01-01,until-pub-date:2021-01-01

https://discussion.theguardian.com/discussion-api/discussion/p/3htd7/topcomments?pageSize=50&page=1&orderBy=newest

https://maps.googleapis.com/maps/api/geocode/json?address=Utrecht&key=YOUR_API_KEY

How to read the URL for the APIs:

- Base API URL: This is the main address where you access the API.
- Endpoints. These specify different data or services. Sometimes, they include required information, like IDs or paths.
- Query (starts with "?"): These are additional parameters, usually optional, that help filter or sort the data you're requesting.
    - When there are multiple parameters in a query, they are joined together with the symbol "&"

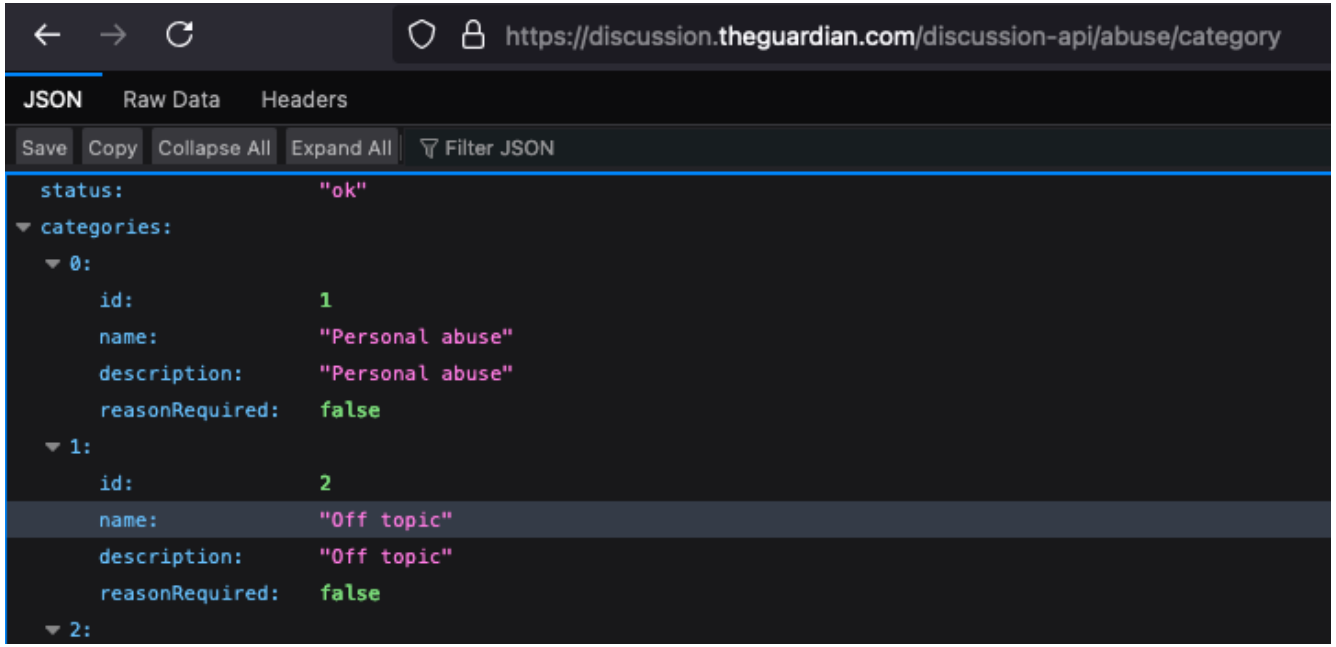
Status response of API

Status code	Meaning
200 OK	Request was successful.
301 Moved Permanently	For SEO purposes when a page has been moved and all link equity should be passed through.
401 Unauthorized	Server requires authentication.
403 Forbidden	Client authenticated but does not have permissions to view resource.
404 Not Found	Page not found because no search results or may be out of stock.
500 Internal Server Error	Server side error. Usually due to bugs and exceptions thrown on the server side code.
503 Server Unavailable	Server side error. Usually due to a platform hosting, overload and maintenance issue.

Example documentation

https://discussion.theguardian.com/discussion-api/abuse/category

Method	Path	Slug	Description	Examples
GET	/abuse/category	GetAllAbuseCategories	Return all abuse categories	/abuse/category
GET	/comment/:id	GetComment	Get a comment	/comment/12500137, /comment/12500137?displayResponses=true, /comment/12500137?displayResponses=true&displayThreaded=true
GET	/comment/:id/context	GetCommentContext	Returns the discussion url, key, and page number on which the comment appears in the discussion. Useful for permalinks.	/comment/12500137/context
POST	/comment/:id/highlight	PostHighlightToComment	Post a highlight to an existing unhighlighted comment	
GET	/comment/:id/permalink	GetCommentPermalink	Redirect to a comment permalink	/comment/1250013/permalink



Example documentation II

https://api.wikimedia.org/feed/v1/wikipedia/en/featured/2024/09/20

Featured content

Discussion Updated 16 June 2023

GET `/feed/v1/wikipedia/{language}/featured/{YYYY}/{MM}/{DD}`

Returns featured content from Wikipedia for a given date. Depending on language availability, the response can include the daily featured article, featured image or media file, list of most read articles, latest news stories, and events from that day in history.

Examples

curl | Python | PHP | JavaScript

```
# Get today's featured content from English Wikipedia
curl https://api.wikimedia.org/feed/v1/wikipedia/en/featured/2024/09/10
```

Parameters

language required path	Language code. For example: `ar` (Arabic), `en` (English), `es` (Spanish). List supported languages.
YYYY required path	Four-digit year
MM required path	Zero-padded month, `01` through `12`
DD required path	Zero-padded day of the month, `01` through `31`

Responses

200	Success **Example** [Show]
400	Error: Invalid parameter **Example** [Show]

Response schema

tfa object	Today's featured article (TFA) for the requested date. Available in 10+ languages. **Properties** [Show]
mostread object	Previous day's most read articles. Available in 300+ languages. **Properties** [Show]
image object	Daily featured image from Wikimedia Commons. Available in English. **Properties** [Show]
news object	Stories from today's news. Available only for the current day in UTC. Available in 15+ languages. **Properties** [Show]
onthisday object	Events that occurred on this day in history. Available in 5+ languages **Properties** [Hide] `text` string — Short summary of the event in plain text `pages` array — Articles related to the event **Properties** [Hide] `type` string — Type of article: • `standard` : Encyclopedia article • `disambiguation` : Page that links to articles covering topics with similar titles • `no-extract` : Article without an extract • `mainpage` : A wiki's homepage

Exercise (in pairs)

Create an API call for Reddit with 100 links containing the query “Utrecht”. Sort hot results first.

Base URL = https://api.reddit.com
Endpoint = /search

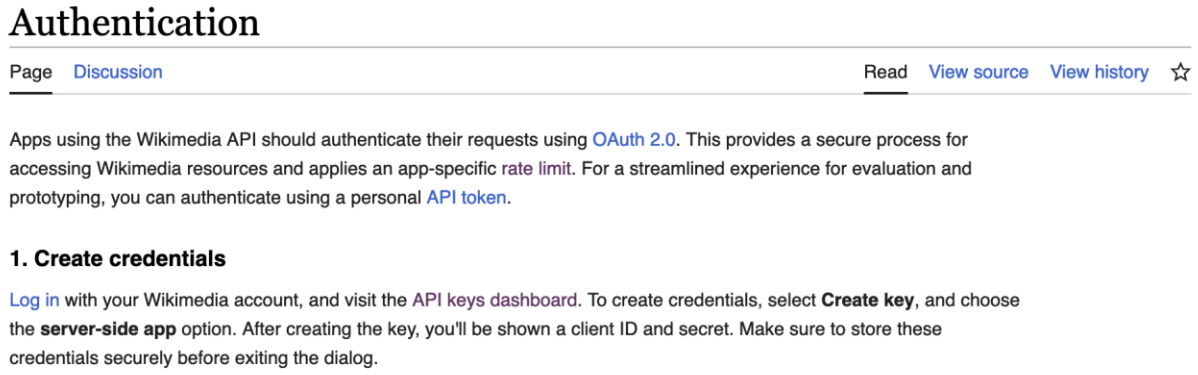
GET **[/r/*subreddit*]/search** read rss support #

Search links page.

This endpoint is a listing.

`after`	fullname of a thing
`before`	fullname of a thing
`category`	a string no longer than 5 characters
`count`	a positive integer (default: 0)
`include_facets`	boolean value
`limit`	the maximum number of items desired (default: 25, maximum: 100)
`q`	a string no longer than 512 characters
`restrict_sr`	boolean value
`show`	(optional) the string `all`
`sort`	one of (`relevance`, `hot`, `top`, `new`, `comments`)
`sr_detail`	(optional) expand subreddits
`t`	one of (`hour`, `day`, `week`, `month`, `year`, `all`)
`type`	(optional) comma-delimited list of result types (`sr`, `link`, `user`)

Authentication and rate limits



Authentication

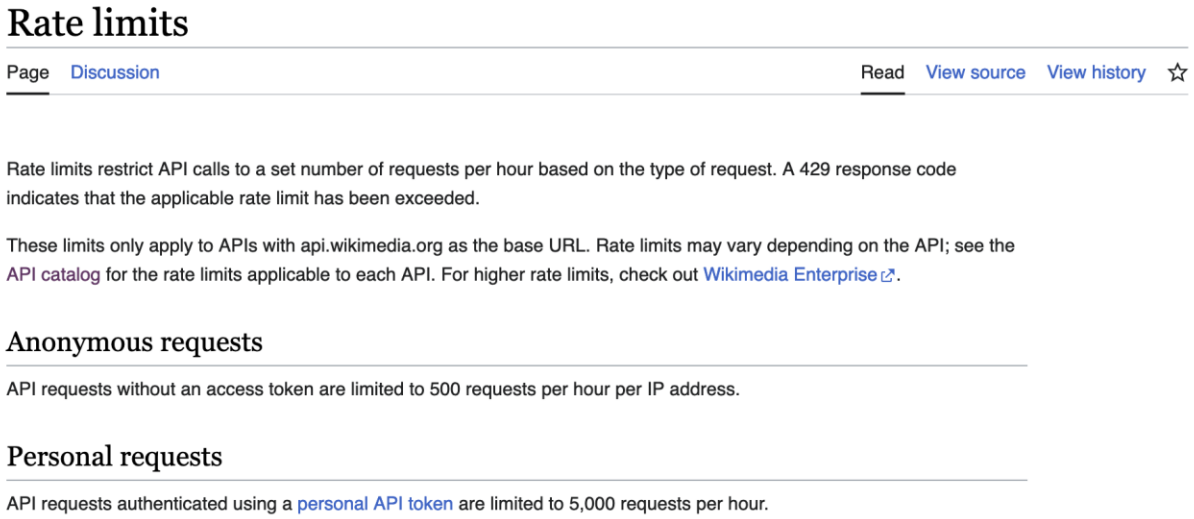
Page Discussion Read View source View history ☆

Apps using the Wikimedia API should authenticate their requests using OAuth 2.0. This provides a secure process for accessing Wikimedia resources and applies an app-specific rate limit. For a streamlined experience for evaluation and prototyping, you can authenticate using a personal API token.

1. Create credentials

Log in with your Wikimedia account, and visit the API keys dashboard. To create credentials, select **Create key**, and choose the **server-side app** option. After creating the key, you'll be shown a client ID and secret. Make sure to store these credentials securely before exiting the dialog.

Allow for different users to have access to different data/services



Rate limits

Page Discussion Read View source View history ☆

Rate limits restrict API calls to a set number of requests per hour based on the type of request. A 429 response code indicates that the applicable rate limit has been exceeded.

These limits only apply to APIs with api.wikimedia.org as the base URL. Rate limits may vary depending on the API; see the API catalog for the rate limits applicable to each API. For higher rate limits, check out Wikimedia Enterprise.

Anonymous requests

API requests without an access token are limited to 500 requests per hour per IP address.

Personal requests

API requests authenticated using a personal API token are limited to 5,000 requests per hour.

Limit the number of requests per minute you can make

WIKIMEDIA

Usually wrappers exist

e.g. API for OpenAI

curl https://api.openai.com/v1/chat/completions \
-H "Authorization: Bearer $OPENAI_API_KEY" \
-H "Content-Type: application/json" \
-d '{ "model": ”çgpt-3.5-turbo", "messages": [{"role": "user", "content": ”Say this is a test"}] }’ç

```python
import os
from openai import OpenAI

client = OpenAI(
    # This is the default and can be omitted
    api_key=os.environ.get("OPENAI_API_KEY"),
)

chat_completion = client.chat.completions.create(
    messages=[
        {
            "role": "user",
            "content": "Say this is a test",
        }
    ],
    model="gpt-3.5-turbo",
)
```

Getting access to an API (e.g. Wikimedia)

1. Log in with your Wikimedia account

To log in to the API Portal, use the same account you use for Wikipedia and other Wikimedia projects. If you're new to Wikimedia (Welcome!), you can create a free account now.

Log in **Create account**

2. Create a personal API token

Once you've logged in, visit API keys to create and manage your API credentials. Select **Create key**, and select the **Personal API token** option. This token is tied to your account. It should only be used by you and should not be published or shared. Remember to store your token in a secure place, like a password manager.

API keys

API key created

Client ID:

5be43a7d72ac8d8091c6eaec

Client secret:

7cff841af8c82341d9b2df33fa6

Access token:

eyJ0eXAiOiJKV1QiLCJhbGci
FjNmVhZWMyODViZmM1ZiIs
YTYxYjIwODY1ZDk4YzRhMz
mE0IiwiaWF0IjoxNzEzOTUyN
XhwIjozMzI3MDg2MTI3OS4x
ldGEud2IraW1lZGlhLm9yZyIs
MCwidW5pdCI6IkhPVVIifSwic
RQUdnA-SlOQDMKmActEcn
XgqNKPO_0H5Hala_50rbYY

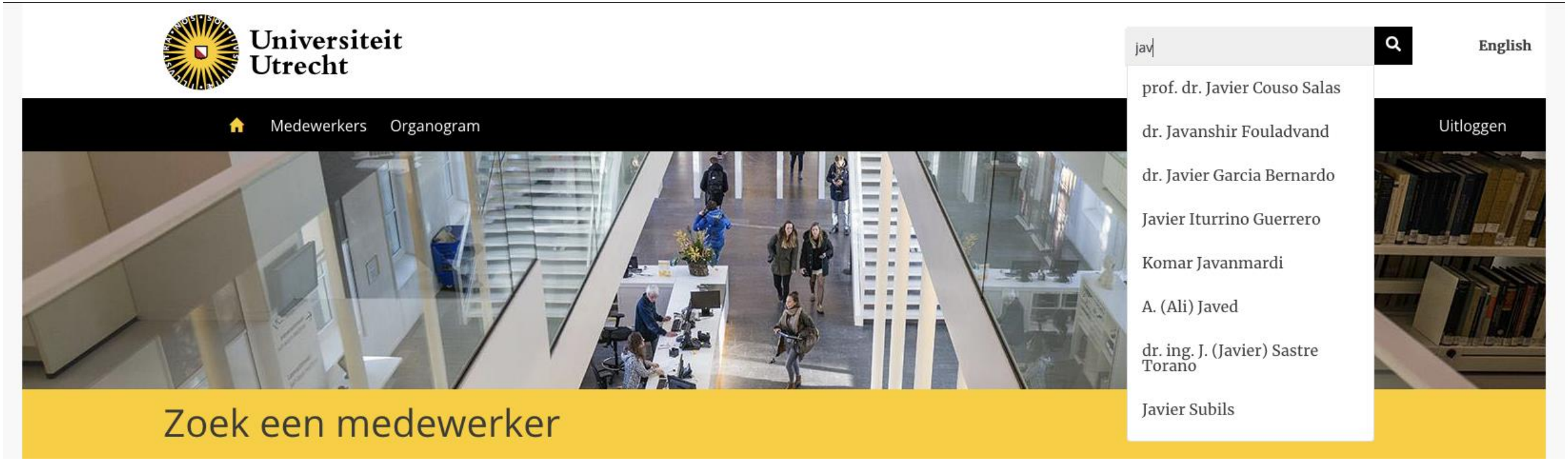
3. Get today's featured article

For your first request, call the featured content endpoint to get today's featured article from English Wikipedia. Use your API token to authenticate the request.

curl | **Python** | PHP | JavaScript

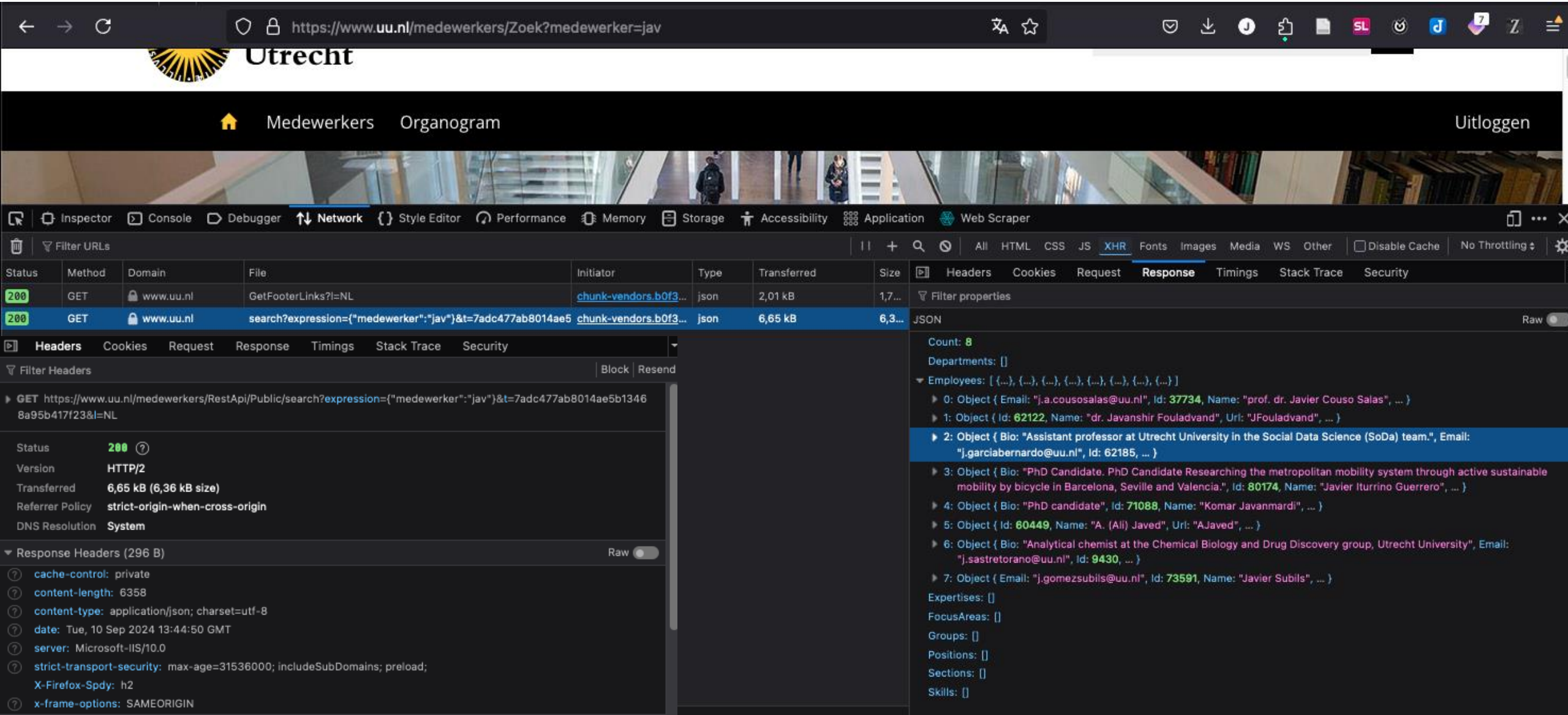
```python
# Python 3
# Get today's featured article from English Wikipedia

import datetime
import requests

today = datetime.datetime.now()
date = today.strftime('%Y/%m/%d')

url = 'https://api.wikimedia.org/feed/v1/wikipedia/en/featured/' + date

headers = {
  'Authorization': 'Bearer YOUR_ACCESS_TOKEN',
  'User-Agent': 'YOUR_APP_NAME (YOUR_EMAIL_OR_CONTACT_PAGE)'
}

response = requests.get(url, headers=headers)
data = response.json()
print(data)
```

Finding private APIs
(be nice and respect robots.txt/ToS/copyright)



Finding private APIs

(be nice and respect robots.txt/ToS/copyright)



API world (before ~2020)

Offering to Bacchus, 1720 (Michel-Ange Houasse)

APIcalypse

(Bruns, 2021)

2015: Linkedin

Reason: Privacy/competition

2015/8: Facebook/Instagram

Reason: Improve privacy

2023: Twitter

Reason: Monetize data access (Musk)

2023: Reddit

Reason: Monetize data access

The Raft of the Medusa, 1818–19 (Théodore Géricault)

Digital Services Act (DSA)

19 October 2022

Article 40, Data access and scrutiny

12. Providers of very large online platforms or of very large online search engines **shall give access without undue delay to data**, including, where technically possible, to real-time data, **provided that the data is publicly accessible in their online interface by researcher**s, including those affiliated to not for profit bodies, organisations and associations, who comply with the conditions set out in paragraph 8, points (b), (c), (d) and (e), and who use the data solely for performing research that contributes to the **detection, identification and understanding of systemic risks** in the Union pursuant to Article 34(1).

Researcher access

Last updated: 1 year ago

As of August 2023, we offer a [Beta] Researcher Access Program in order to meet our legal requirements under the European Union's Digital Services Act (DSA). Specifically, Article 40(12) of the DSA requires that we make available to qualified researchers, upon application, data that is publicly accessible on LinkedIn, so long as the following conditions are satisfied:

- Researchers must be independent from commercial interests.
- The researcher's application must disclose the funding of the research

Digital Services Act (DSA)

19 October 2022

EN English

Search Search

PRESS RELEASE | 30 April 2024 | Brussels | 6 min read

Commission opens formal proceedings against Facebook and Instagram under the Digital Services Act

PRESS RELEASE | 19 February 2024 | Brussels | 3 min read

Commission opens formal proceedings against TikTok under the Digital Services Act

PRESS RELEASE | 18 December 2023 | Brussels | 4 min read

Commission opens formal proceedings against X under the Digital Services Act

PRESS RELEASE | 14 March 2024 | Brussels | 3 min read

Commission opens formal proceedings against AliExpress under the Digital Services Act

Mechanism Description	Interface	Who has access?	Application required	Data Dictionary /Docu...
Apple Ad Repository and API	Searchable web interface and API	Public	No	Yes
Booking Ads Repository and API	Searchable web interface and API	Public	No	Yes
Google Ads Transparency Center	Searchable web interface and API	Public	No	Yes
LinkedIn Ad Library	Searchable web interface and API	Public	Yes	Yes
Meta Ad Library	Searchable web interface and API	Public	No	Yes
Microsoft Ad Library	Searchable web interface and API	Public	No	Yes
Pinterest Ads Repository	Searchable web interface and API	Public	No	No
Snap Ads Gallery	Searchable web interface	Public	No	No
Snap Political Ads Library	CSV Files	Public	No	Yes
TikTok Ad Library	Searchable web interface and API	Public	Yes	Yes
X (formerly Twitter) Ads Repository	Searchable web interface and API	Public	No	Yes

Mechanism Description	Interface	Who has access?	Application required	Data Dictionary /Docu...
AliExpress Open Research & Transparency	The mechanism description URL suggest...	Academic and civil society researchers as describ...	Yes	No
Booking.com Scraping Provision1	API "as applicable" (suggesting other met...	Anyone scraping for non-commercial purposes	No	No
Bing Qualified Researcher Program	API "as applicable" (suggesting other met...	Academic and civil society researchers as describ...	Yes	No
Google Request Records	Varies, see below:	Researchers affiliated with EU-based organizations	Yes	No
LinkedIn Researcher Access	API "as applicable" (suggesting other met...	Academic and civil society researchers as describ...	Yes	No
Meta Content Library and API	Searchable user interface and API provid...	To be eligible for product access, researchers mu...	Yes	Yes
Pinterest Researchers Intake		“If you’re a researcher”	Yes	No
Reddit Researcher Access Request	Commercial API	Researchers accessing data for non- commercial ...	Yes	Yes
Snap Researcher Data Access		Requests are "in accordance with the Digital Servi...	Yes	No
TikTok Research API	API	Researchers from US and Europe	Yes	Yes
Wikipedia Tools	Scraping and a set of APIs	Public	No	Yes
X (formerly Twitter) API	Commercial API	Different levels of access on the basis of fees and...	Yes	Yes
YouTube Researcher Program	Commercial API	Must be affiliated with an “eligible academic instit...	Yes	Yes

Advantages and Disadvantages of APIs

Advantages for the company (providing the API):

- Track data usage
- Security and authentication: The company can control who gets access to their data by requiring API keys and user authentication.
- Rate limits: The company can limit how many requests are made to the API (e.g., a maximum number of requests per minute) to prevent abuse and ensure their servers aren't overwhelmed.

Advantages for you (the user of the API):

- Ease of use: they are documented and often have Python/R packages
- No legal concerns (as long as you follow the Terms of Service)

Disadvantages for you:

- Limited data access: Only public data, or data where individuals cannot be identified
- Data availability: The data you want may not be available (e.g. Facebook) or be very expensive (e.g. Twitter)

Collaboration with companies

Meta/Facebook case

SOCIAL SCIENCE ONE

Hosted by Harvard's Institute for Quantitative Social Science

BLOG CONTACT US

Building Industry-Academic Partnerships

Projects must be focused on the effects of social media on democracy and elections.

RFP for URL Shares

This is a codebook for data on the demographics of people who viewed, shared, and otherwise interacted with web pages (URLs) shared on Facebook. The data has about 68 million URLs, over 3.1 trillion rows, and over 71 trillion cell values. It results from a collaboration between Facebook and Social Science One (at IQSS at Harvard), originally prepared for Social Science One grantees and describes the "full" URLs dataset, including its scope, structure, and fields. This is version 10 of the codebook and data (released 4/13/2023), first described by Gary King and Nathaniel Persily at https://socialscience.one/blog/update-social-science-one (2023).

Aggregated data available directly

FACEBOOK

Accessible data includes posts shared to and information about Pages, groups and events, as well as a subset of public profiles that have a verified badge or 25,000 or more followers.

INSTAGRAM

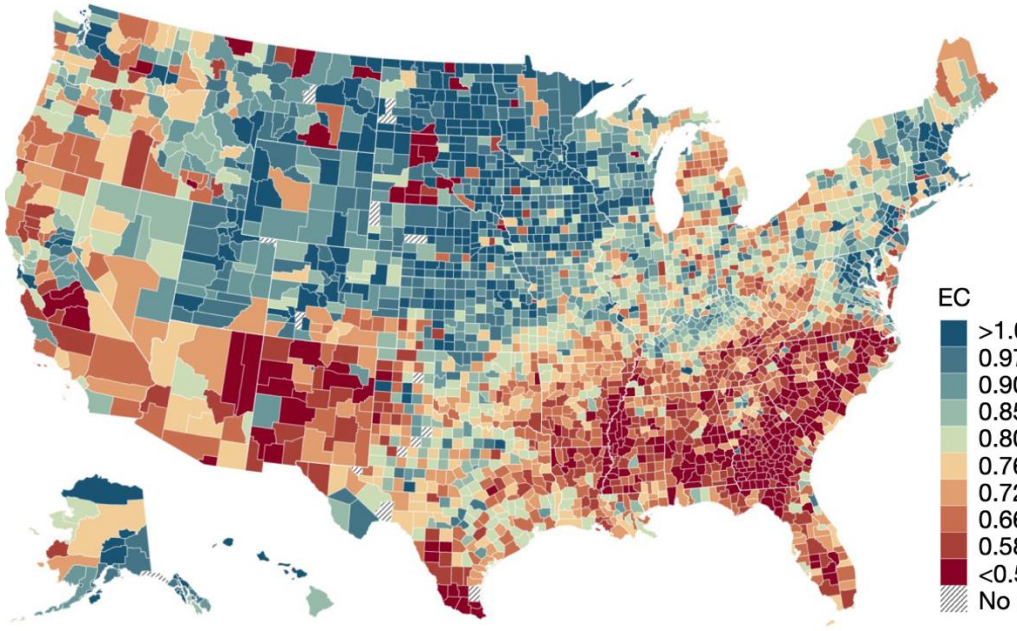
Accessible data includes posts shared by and information about business and creator accounts, as well as a subset of personal accounts that have been set to public and have a verified badge or 25,000 or more followers.

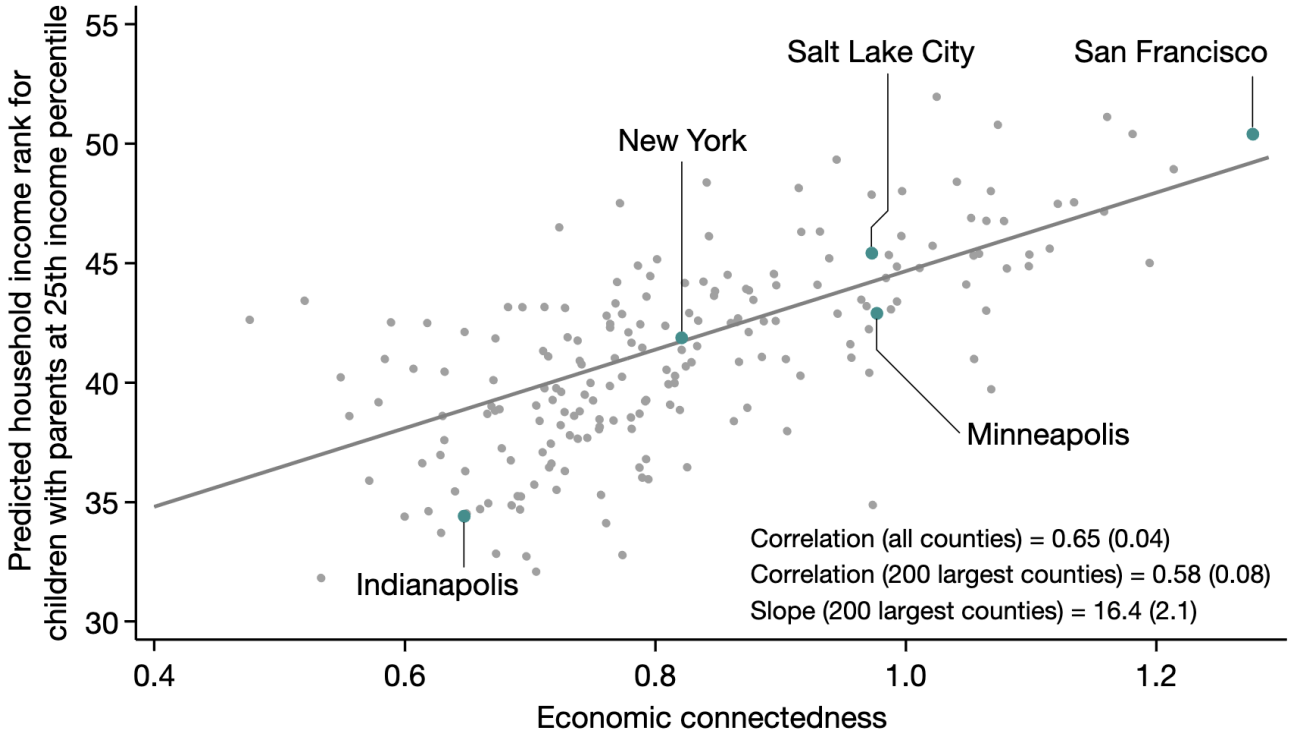
GEOGRAPHICAL DATA

Data is available for most countries and territories but excluded from countries where Meta is still evaluating legal and compliance requirements.

POST VIEW DATA

The post views metric indicates the number of times a post or reel was displayed on screen.





Chetty et al, 2022, Natur

15 min break

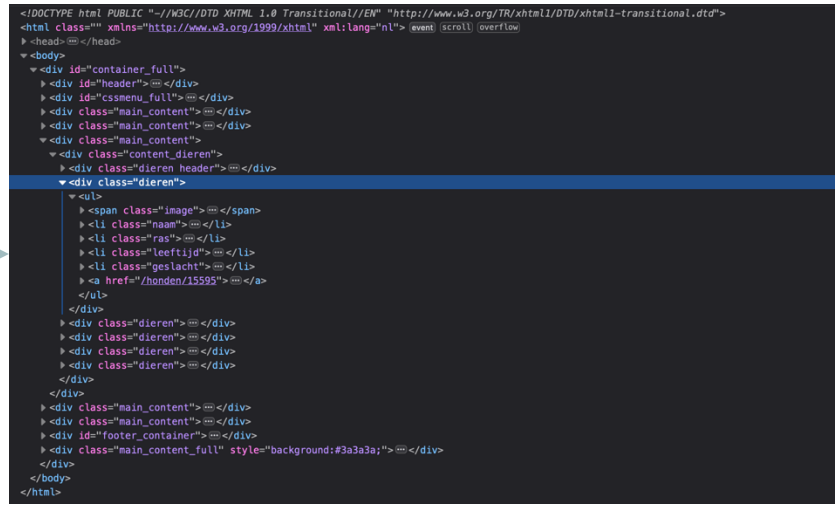
Web Scraping

What is web scraping?

Extracting data from the web in an automatic manner

	Naam	Ras	Leeftijd	Geslacht	
	Tibbe	Kruising doodle	Volwassen	Reu	Beschikbaar
	Pepper	Boerboel	Volwassen	Reu	Beschikbaar
	Pepper	Amerikaanse Bulldog	Volwassen	Teef	Beschikbaar
	Rocky	jack russel terriër	Volwassen	Reu	Geplaatst
	Dexx	Kruising	Volwassen	Reu	Beschikbaar
	Poppy	Kruising	Volwassen	Teef	Beschikbaar

Start with a website you can legally scrape



1. Download the HTML code of the website

2. Parse the HTML

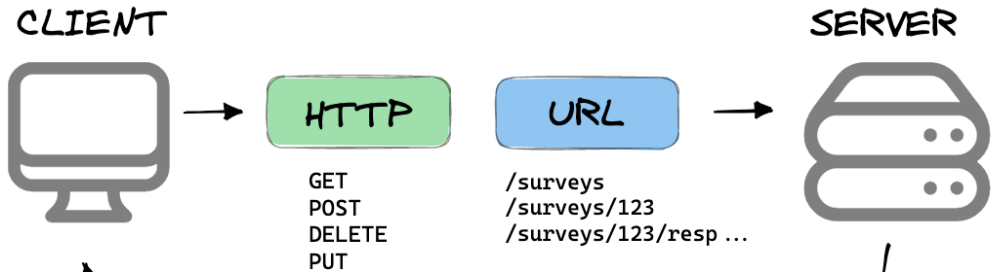
American Political Science Review Vol. 107, No. 2 May 2013
doi:10.1017/S0003055413000014

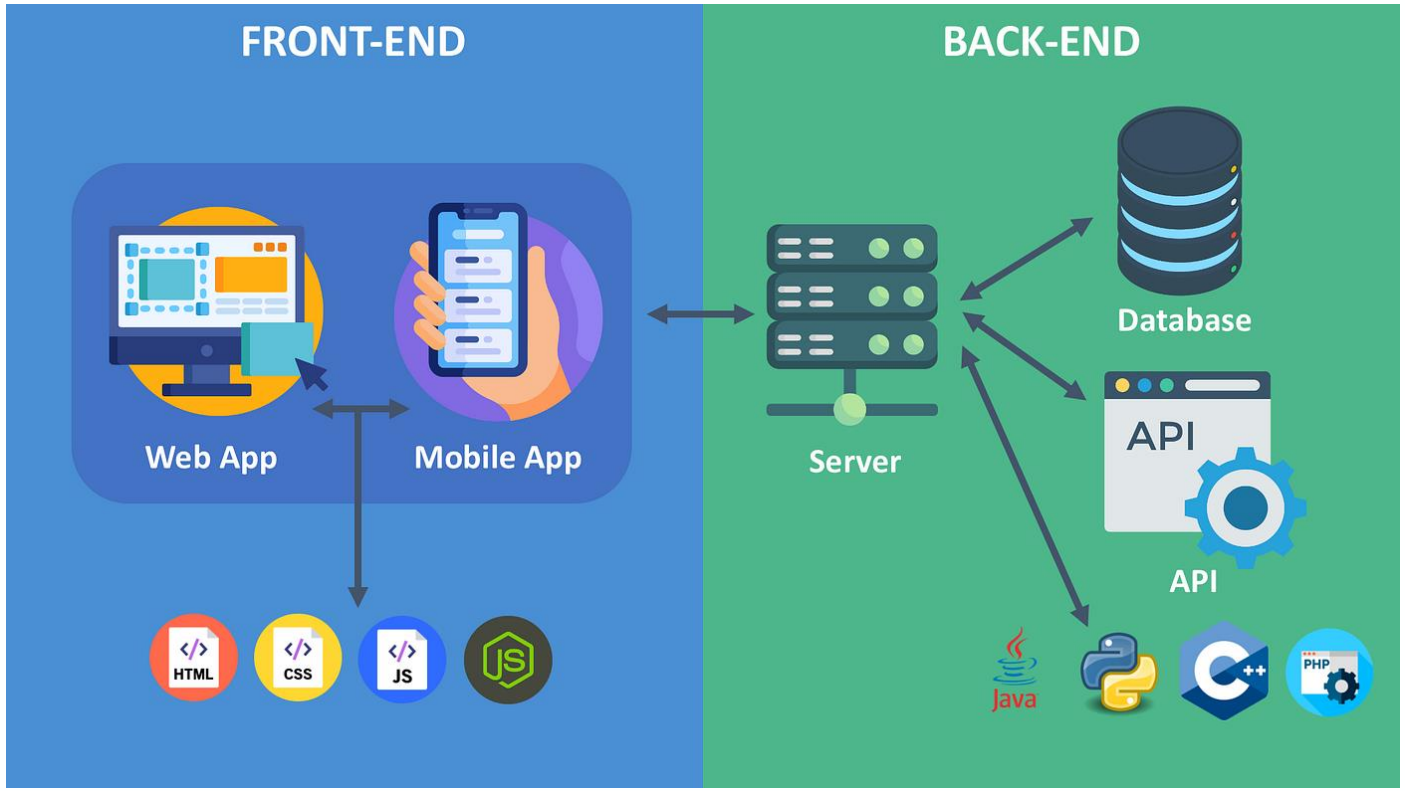
How Censorship in China Allows Government Criticism but Silences Collective Expression

GARY KING *Harvard University*
JENNIFER PAN *Harvard University*
MARGARET E. ROBERTS *Harvard University*

We offer the first large scale, multiple source analysis of the outcome of what may be the most extensive effort to selectively censor human expression ever implemented. To do this, we have devised a system to locate, download, and analyze the content of millions of social media posts originating from nearly 1,400 different social media services all over China before the Chinese government is able to find, evaluate, and censor (i.e., remove from the Internet) the subset they deem objectionable. Using modern computer-assisted text analytic methods that we adapt to and validate in the Chinese language, we compare the substantive content of posts censored to those not censored over time in each of 85 topic areas. Contrary to previous understandings, posts with negative, even vitriolic, criticism of the state, its leaders, and its policies are not more likely to be censored. Instead, we show that the censorship program is aimed at curtailing collective action by silencing comments that represent, reinforce, or spur social mobilization, regardless of content. Censorship is oriented toward attempting to forestall collective activities that are occurring now or may occur in the future—and, as such, seem to clearly expose government intent.

Web pages





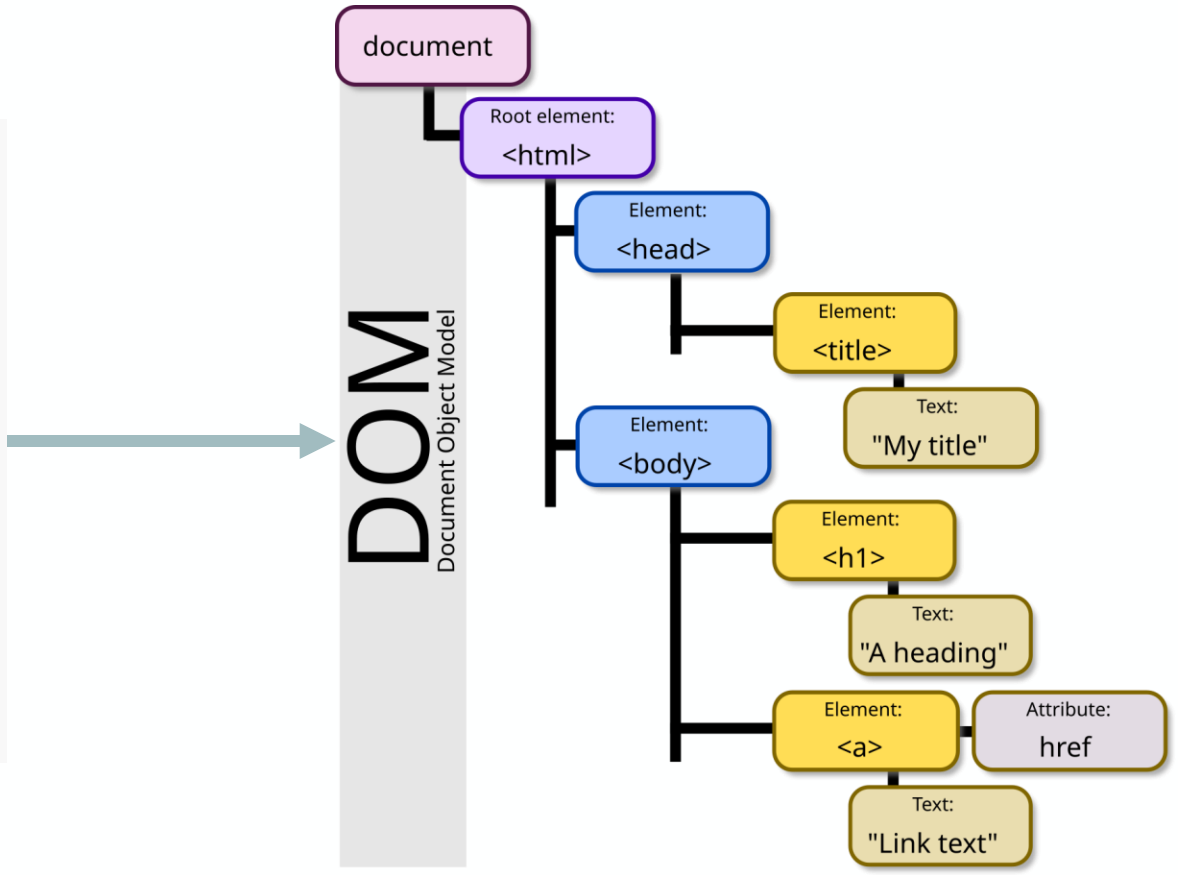
- **HTML** (HyperText Markup Language): The skeleton of a website, containing the sections and text.
- **CSS** (Cascading Style Sheets): Makes it pretty by adding styles and colors.
- **JS** (JavaScript): Brings complex interactivity and logic to the website

https://medium.com/@adityagaba1322/streamlining-backend-frontend-integration-a-quick-guide-145eca3cca05

Document Object Model (DOM)

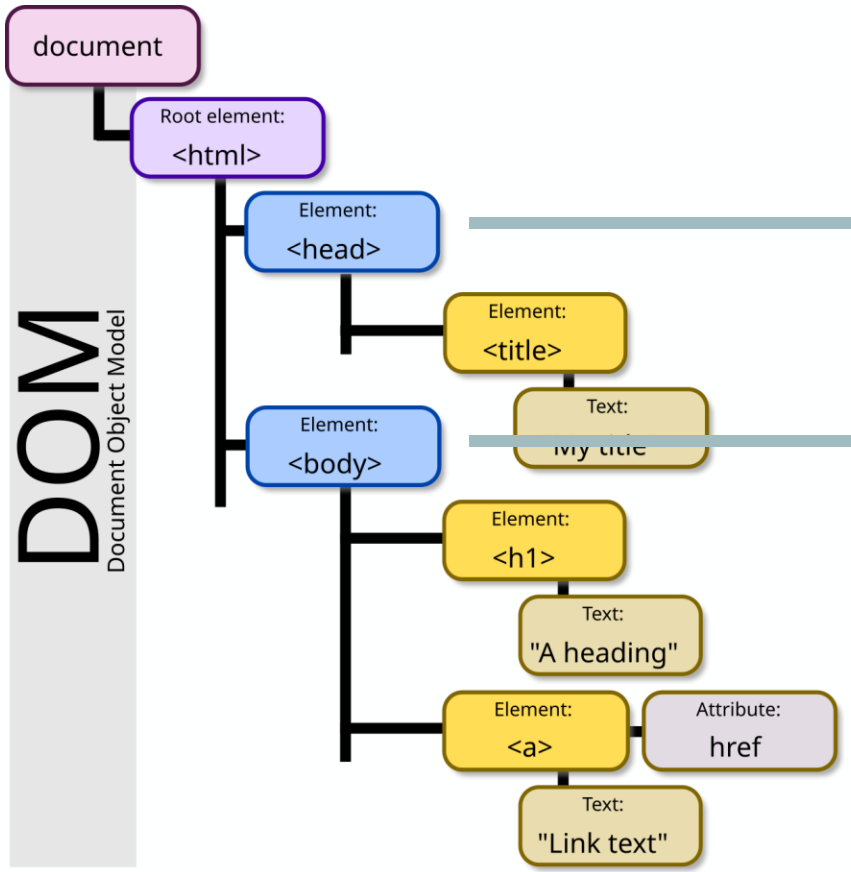
Represents the HTML code as a tree

```
<html>
  <head>
    <title>My Website</title>
  </head>
  <body>
    <h1>Welcome</h1>
    <p>This is my website.</p>
  </body>
</html>
```



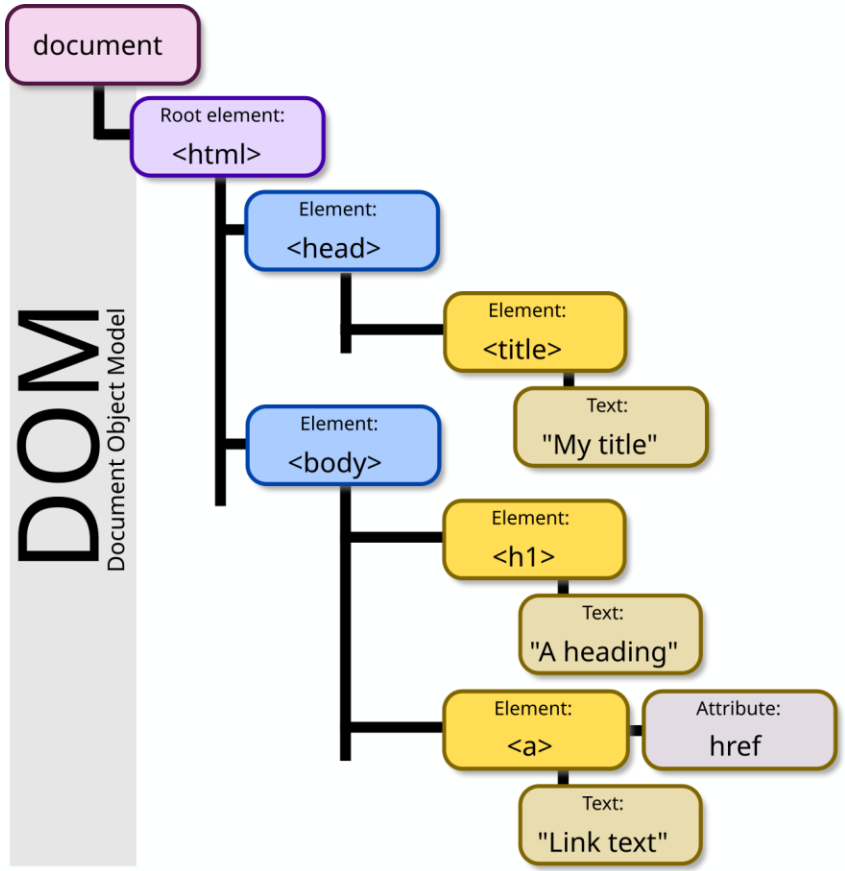
Birger Eriksson, Wikipedia

Document Object Model (DOM)



Metadata, title, load styles (css) and scripts (js)

Content



Elements (tags):

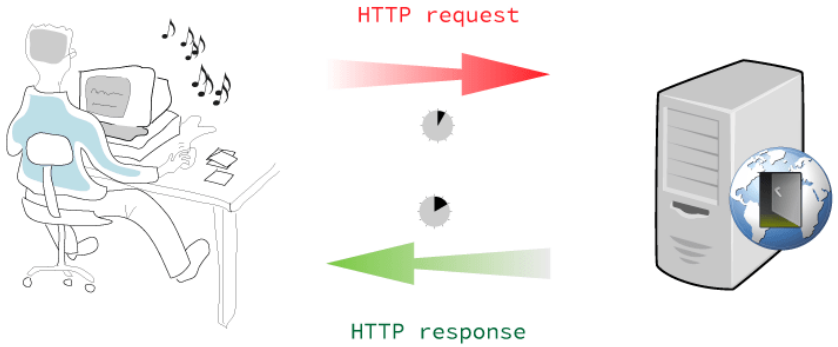
- div: basic container
- a: link to url
- p: paragraph
- h1-h6: titles
- img: image
- table: table

Attributes:

- id: unique of the element
- class: set the style/interactivity for multiple elements
- href: url
- …

https://www.tutorialrepublic.com/html-tutorial/html-elements.php?pStoreID=techsoup%27[0]

Scraping static websites



https://about.gitlab.com/blog/2016/06/03/ssg-overview-gitlab-pages-part-1-dynamic-x-static/

	Naam	Ras	Leeftijd	Geslacht	
	Tibbe	Kruising doodle	Volwassen	Reu	Beschikbaar
	Pepper	Boerboel	Volwassen	Reu	Beschikbaar
	Pepper	Amerikaanse Bulldog	Volwassen	Teef	Beschikbaar
	Rocky	jack russel terriër	Volwassen	Reu	Geplaatst
	Dexx	Kruising	Volwassen	Reu	Beschikbaar
	Poppy	Kruising	Volwassen	Teef	Beschikbaar

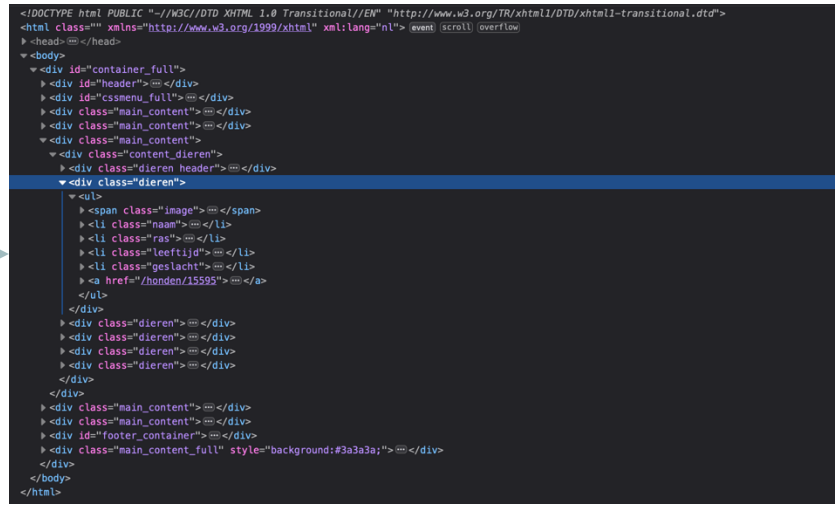
https://www.dierenasielutrecht.nl/honden/

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
<html class="" xmlns="http://www.w3.org/1999/
▶ <head>…</head>
▼ <body>
  ▼ <div id="container_full">
    ▶ <div id="header">…</div>
    ▶ <div id="cssmenu_full">…</div>
    ▶ <div class="main_content">…</div>
    ▶ <div class="main_content">…</div>
    ▼ <div class="main_content">
      ▼ <div class="content_dieren">
        ▶ <div class="dieren header">…</div>
        ▼ <div class="dieren">
          ▼ <ul>
            ▶ <span class="image">…</span>
            ▶ <li class="naam">…</li>
            ▶ <li class="ras">…</li>
            ▶ <li class="leeftijd">…</li>
            ▶ <li class="geslacht">…</li>
            ▶ <a href="/honden/15595">…</a>
            </ul>
          </div>
        ▶ <div class="dieren">…</div>
        ▶ <div class="dieren">…</div>
        ▶ <div class="dieren">…</div>
        ▶ <div class="dieren">…</div>
        </div>
      </div>
    ▶ <div class="main_content">…</div>
    ▶ <div class="main_content">…</div>
    ▶ <div id="footer_container">…</div>
    ▶ <div class="main_content_full" style="background:#3a3a3a;">…</div>
    </div>
  </body>
</html>
```

	Naam	Ras	Leeftijd	Geslacht	
	Tibbe	Kruising doodle	Volwassen	Reu	Beschikbaar
	Pepper	Boerboel	Volwassen	Reu	Beschikbaar

```
▼ <div class="dieren">
  ▼ <ul>
    ▶ <span class="image">…</span>
    ▼ <li class="naam">
        <span>Naam:</span>
        Tibbe
      </li>
    ▼ <li class="ras">
        <span>Ras:</span>
        Kruising doodle
      </li>
    ▼ <li class="leeftijd">
        <span>Leeftijd:</span>
        Volwassen
      </li>
    ▼ <li class="geslacht">
        <span>Geslacht:</span>
        Reu
      </li>
    ▼ <a href="/honden/15680">
      ▶ <li class="status beschikbaar">…</li>
      </a>
    </ul>
  </div>
```

Start with a website you can legally scrape



1. Download the HTML code of the website

2. Parse the HTML

1. Download the HTML code of the website (usually easy)

```
import requests
r = requests.get("https://www.dierenasielutrecht.nl/honden/")
print(r.status_code)
print(r.text)
```

```
200
<!DOCTYPE HTML PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="nl">
<head>
        <title>Honden - Dierenasiel Utrecht</title>
                <meta name="author" content="Dierenasiel, Utrecht" />
        <meta http-equiv="content-type" content="text/html; charset=utf-8" />
        <meta http-equiv="content-script-type" content="text/javascript" />
        <meta http-equiv="content-style-type" content="text/css" />
        <meta name="viewport" content="width=device-width, initial-scale=1.0" />
```

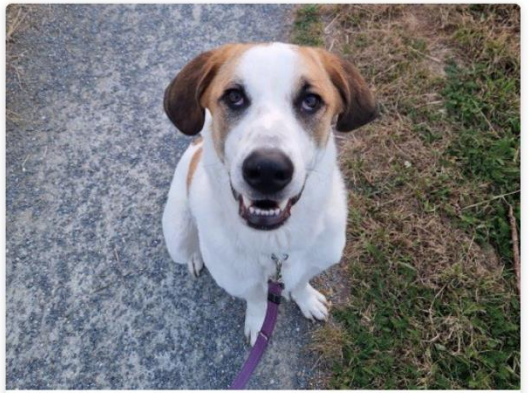
2. Parse the HTML (prone to errors)

```
import bs4
html = bs4.BeautifulSoup(r.text)
html.find_all("li", attrs={"class": "naam"})
```

```
[<li class="naam">Naam</li>,
 <li class="naam"><span>Naam:</span>Tibbe</li>,
 <li class="naam"><span>Naam:</span>Pepper </li>,
 <li class="naam"><span>Naam:</span>Pepper</li>,
 <li class="naam"><span>Naam:</span>Dexx</li>,
 <li class="naam"><span>Naam:</span>Poppy</li>]
```

Exercise (in pairs)

What HTML element contains the date when the dog was added?
What element/attribute would you need to select to extract the image?



Milo

IJMUIDEN 18-09-2024

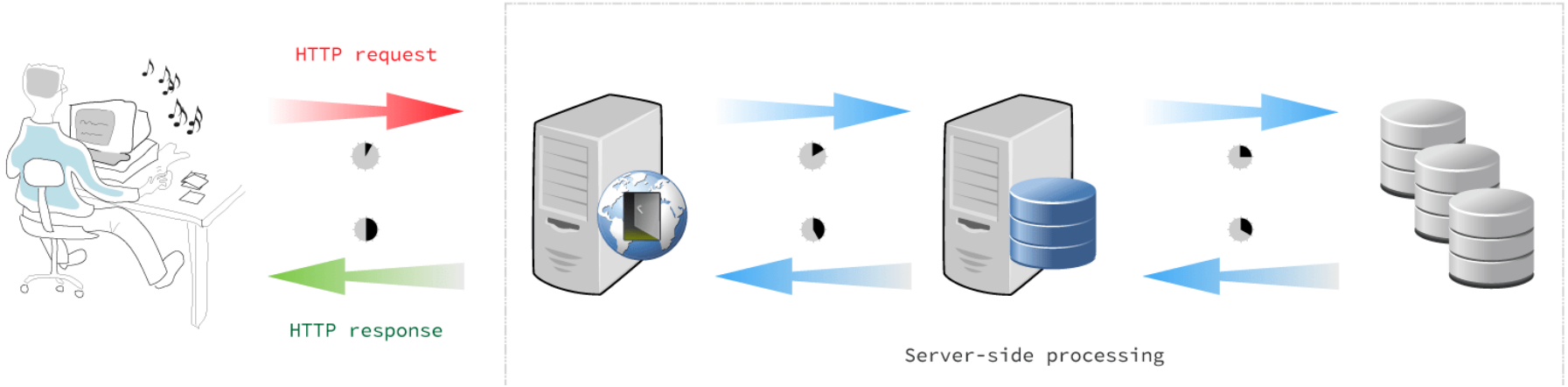
```
<a class="animal-card-grid" target="_blank" href="/plaatsprofiel/189022-adoptiehond-zoekt-baasje-griekse-berghondbeagle-ijmuiden/"> event grid
  <div class="card-image">
    <img class="img-responsive img-fluid w-100 h-100" src="https://verhuisdieren.nl/images/189022/462608/conversions/blob-thumbnail.jpg?v=1726556916" alt="Milo">
    event
  </div>
  <div>
    <div class="card-header">
      <div class="align-items-center">
        <h3 class="font-size-subtitle text-dark">Milo</h3>
      </div>
    </div>
    <div class="card-body">
      <div class="d-flex justify-content-between align-items-center"> flex
        <div>
          <label class="font-family-system">IJmuiden</label>
          <span class="text-truncate text-gray-light">18-09-2024</span>
        </div>
        <div class="text-gray-light small"></div>
```

https://verhuisdieren.nl/honden/adoptiehonden/

Scraping dynamic Websites

Examples:

- Infinite scrolls: Many social media platforms (like Facebook, Instagram, or Twitter) continuously load new content as you scroll down the page without needing to click “next.”
- Websites that update or change part of the page without reloading when you interact with them



https://about.gitlab.com/blog/2016/06/03/ssg-overview-gitlab-pages-part-1-dynamic-x-static/


Your scraper
OHMAGIF.COM

How to scrape websites practically?

Programming:

- Static websites: *requests* + beautifulsoup (*bs4*)
- Dynamic websites: *selenium*

Web browser extensions (this afternoon)**:**

- Click on elements
- The software figures it out by itself
- Not so flexible

Advantages and Disadvantages of Scraping

Advantages for the company (the website being scraped):

- No need to provide an API

Disadvantage for the company:

- Loss of control over data

Advantages for you (the person scraping the data):

- Access to public data: You can collect and analyze any data that is publicly available on the website, even if the site doesn't offer an API for it.

Disadvantages for you:

- Limited to public data: You can only scrape data that is publicly visible-–not behind logins/paywalls
- It is usually hard!
  - Technical challenges: Especially if the website actively tries to prevent scraping (e.g. captchas) or if the website's structure changes.
  - Inconsistent data: Scraped data is often messy/inconsistent.
- Legal challenges: Many websites explicitly forbid web scraping. It is complex legally.

Legality of APIs/web scraping

Different levels (specific for web scraping/APIs)

robots.txt

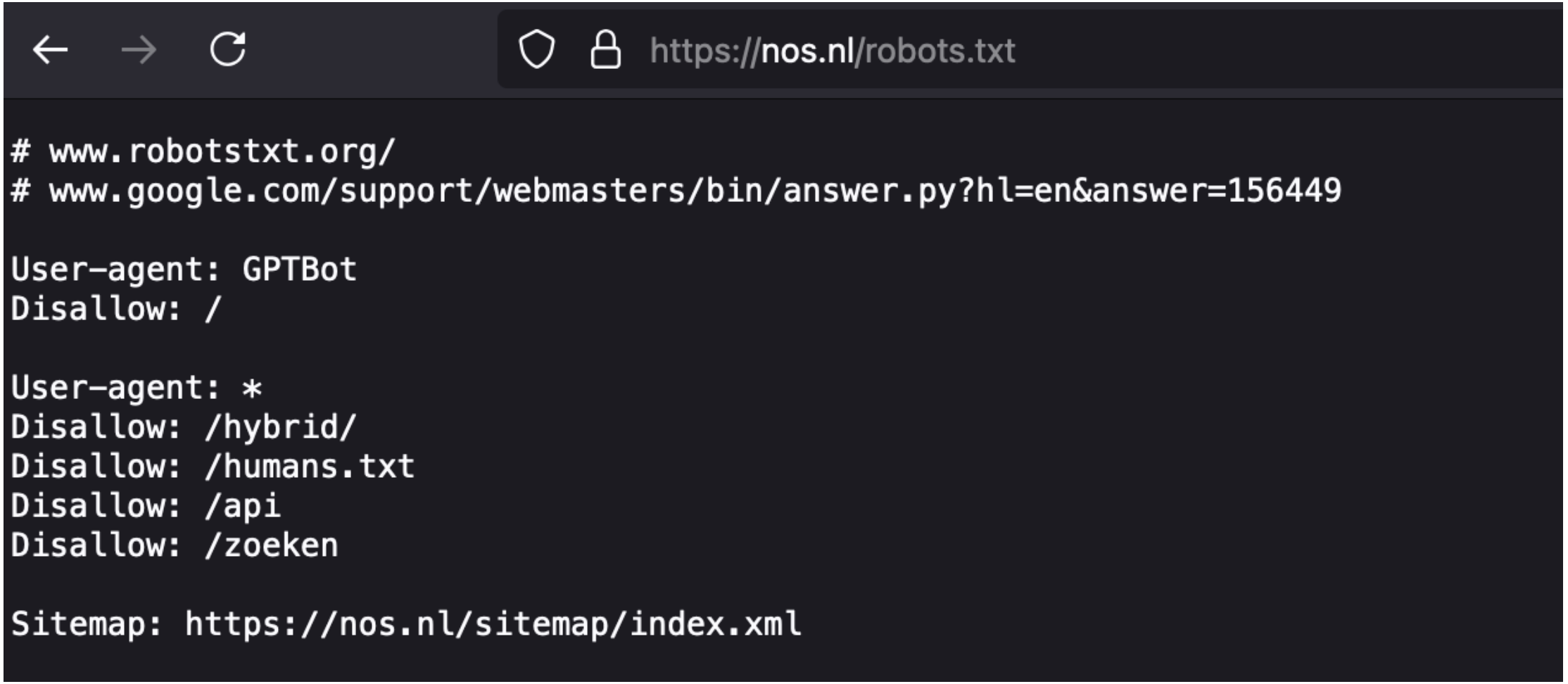
- Used to communicate what parts of a website should or should not be accessed
- Not legally binding, but please respect them!

Terms of Service

- Legal agreements that define how a user or a scraper are allowed to interact with a website
- Only enforceable if the user clicks "I agree" (e.g. by creating an account, or through a pop-up)
- Can have serious legal consequences, especially if the user bypasses authentication to download data (e.g. created fake account to access data, scraped data in bulk from an institutional account, etc).
- Some companies (LinkedIn, Meta) have already won legal battles against scraping for commercial purposes

Aaron Swartz

Robots.txt


https://nos.nl/robots.txt
www.robotstxt.org/
www.google.com/support/webmasters/bin/answer.py?hl=en&answer=156449
User-agent: GPTBot
Disallow: /
User-agent: *
Disallow: /hybrid/
Disallow: /humans.txt
Disallow: /api
Disallow: /zoeken
Sitemap: https://nos.nl/sitemap/index.xml

Terms of Service

App/Service	Word Count	How many minutes to read? (240 wpm)
Microsoft	15,260	63.5
Spotify	8,600	35.8
Niantic (Pokemon Go)	8,466	35.2
TikTok	7,459	31.4
Apple (Media Services)	7,314	30.5
Zoom	6,891	28.7
Tinder	6,215	25.9
Slack	5,782	24.1
Uber	5,658	23.6
Twitter	5,633	23.5

https://www.visualcapitalist.com/terms-of-service-visualizing-the-length-of-internet-agreements/

More regulations (EU)

Database Directive (Directive 96/9/EC)
- Databases are protected products
- Non-commercial scientific research is exempted (with caveats)

Copyright law
- The copyright of created materials can lie with the user (e.g. Reddit) or the platform (e..g TikTok)
- Personal data cannot be protected by copyright

Digital Copyright Directive (Directive 2019/790)
- Exemptions for text and data mining: Research are allowed to mine public data (like Reddit posts) for non-commercial research purposes, even without permission from the rightsholder, but only if the data is lawfully accessed.

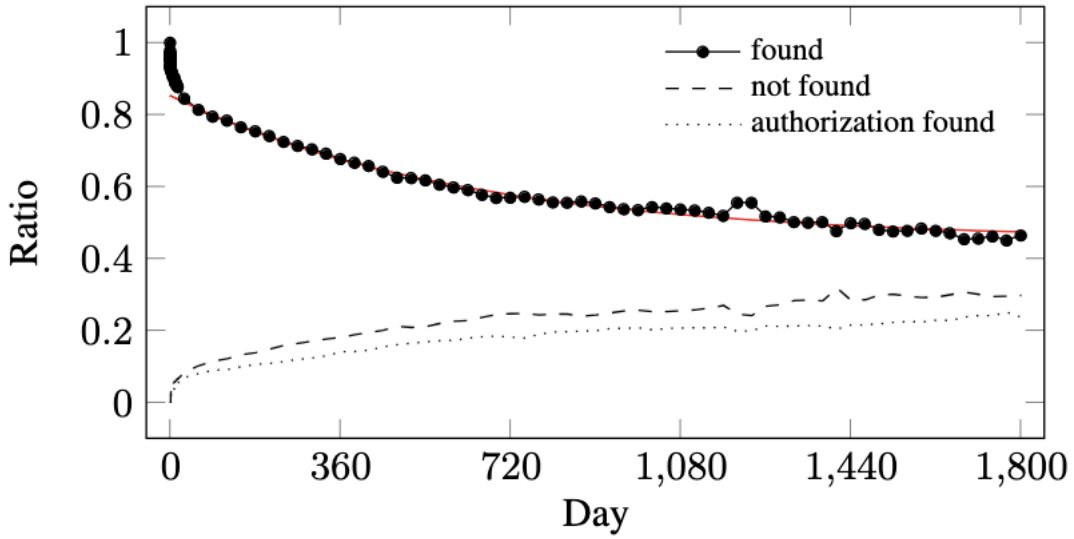
Privacy regulations: GDPR

Last week:

- GDPR enables data donation (rights of access and data portability)
- *Personal data:* Information relating to an identified or identifiable natural person.
- *Data controller:* The person or organization responsible for processing personal data.

If you collect personal data, you become a data controller!

- *Right to Rectification (Article 16):* Individuals can request that inaccurate or incomplete personal data be corrected.
- *Right to Erasure (Article 17):* Individuals can request the deletion of their data.



Pfeffer et al., 2023

GDPR Principles in Data Collection for Research

Lawfulness, Fairness, and Transparency

- This is key! Collect data with a legal basis (explicit *informed consent* OR *legitimate interest* that outweighs any privacy risk to individuals)

Purpose limitation, data minimization and storage limitation

- Data must be collected for specific, explicit and legitimate purposes
- Only collect data that is necessary
- Only keep personal data as long as necessary

Accuracy, security and accountability

- Ensure data is accurate and up to date: provide data subjects the right to access, rectify, and delete data
- Protect personal data

Sensitive personal data (extra protected**):** Genetic, biometric, health, racial or ethnic origin, political opinions, religious or ideological convictions, trade union memberships.

Anonymization and Pseudonymization

GDPR only applies to personal data.

Anonymization means data is processed so individuals cannot be identified, even with additional information.

- *Example:* Removing names, birthdates, and any other identifiable information from a dataset, leaving only aggregated statistics.
- Anonymized data is not personal data and GDPR does not apply

Pseudonymization involves replacing identifying information (e.g., names) with fake identifiers, but the data can still be linked back to individuals with additional information.

- Pseudoanonymized data is considered personal data.
- You should still do it when full anonymization is not possible (it reduces risks)

It is your responsibility as data collector to handle the data appropriately

Personal data

Person	Country of Origin	Grade
Emma Kelswick	Netherlands	92
Jonah Merrick	Netherlands	88
Lila Ravenswood	Netherlands	95
Ethan Broderick	Netherlands	89
Ava Thornhill	Netherlands	94
Lucas Fairmont	Germany	87
Nora Windrow	South Africa	90
Caleb Durnford	Netherlands	85
Isla Hargrave	Netherlands	91
Mason Larkspur	Netherlands	93

Is this personal data?

Person	Country of Origin	Grade
Person 1	Netherlands	92
Person 2	Netherlands	88
Person 3	Netherlands	95
Person 4	Netherlands	89
Person 5	Netherlands	94
Person 6	Germany	87
Person 7	South Africa	90
Person 8	Netherlands	85
Person 9	Netherlands	91
Person 10	Netherlands	93

Is this personal data?

Country of Origin	Grade
Netherlands	92
Netherlands	88
Netherlands	95
Netherlands	89
Netherlands	94
Germany	87
South Africa	90
Netherlands	85
Netherlands	91
Netherlands	93

Clearview AI gets another EU fine

Share

By Pieter Cranenbroek, Editor at LinkedIn News

Updated 6 days ago

The Dutch Data Protection Authority has fined Clearview AI €30.5m for violating European privacy law. According to the Dutch authority, the facial recognition company **illegally maintains a database** of billions of photos, which were taken from the internet without the knowledge or consent of the people in question. If the US company does not change the way it operates, a penalty of up to €5m could be added on top of the fine. Clearview AI said the decision was **"unlawful"** as it does not have a place of business or customers in the EU. The data watchdog is also investigating whether the company's directors can be held **personally liable** for failing to take action despite previous fines from other authorities.

- Clearview AI has now been **fined a total of €90.5m in the EU** as authorities in France, Italy and Greece previously found the company in breach of data protection regulations.

Ethics?

https://www.transcend.org/tms/2020/07/right-moral-ethical-vs-legal/

Let's talk about it in 3 weeks!

Different definition of what is important (morality? consequences?)

Three guiding principles:

- **Respect for Persons:** Respect autonomy: receive informed consent if possible.
- **Beneficence and respect for public interest**: Do no harm // maximize the benefit/risk ratio
- **Justice**: Ensure the risks and benefits of research are distributed fairly

RESEARCH ARTICLE | PSYCHOLOGICAL AND COGNITIVE SCIENCES |

Experimental evidence of massive-scale emotional contagion through social networks

Adam D. I. Kramer, Jamie E. Guillory, and Jeffrey T. Hancock Authors Info & Affiliations

Edited by Susan T. Fiske, Princeton University, Princeton, NJ, and approved March 25, 2014 (received for review October 23, 2013)

June 2, 2014 | 111 (24) 8788-8790 | https://doi.org/10.1073/pnas.1320040111

When positive expressions were reduced, people produced fewer positive posts and more negative posts

The authors noted in their paper, "[The work] was consistent with Facebook's Data Use Policy, to which all users agree prior to creating an account on Facebook, constituting informed consent for this research."

Exercise (in pairs)

We are interested in detecting how implicit racism/sexism/classism on LinkedIn leads to inequalities in job acquisition.

Below are the ToS of LinedIn.

- Could we create fake accounts to scrape photos and interactions between people?
- Could we scrape the information publicly available on Google?
- Could we use a plugin that downloads the data that we see in our personal account?

What would we need to do to be compliant with GDPR?

Customer agrees that it will not:

· Except as expressly authorized by LinkedIn in writing, use any automated means or form of scraping or data extraction to access, modify, download, query or otherwise collect information from LinkedIn's websites; or

	API Data	Web Scraping	Data Donations	Tracking (plugins)
User- vs. platform centrality				
Definition				
Time frame of collected data				
Types of data				
Robots.txt applies				
Terms of Service apply				
GDPR applies				
Consent of participants				
Potential for reactivity/ social desirability biases				
Level of gathered content				
Privacy risks in the collection of personally identifiable information				
Main advantages				
Main challenges				

Ohme et al, 2024, Communication Methods and Measures

	API Data	Web Scraping	Data Donations	Tracking (plugins)
User- vs. platform centrality	Platform	Platform	User	User
Definition	Official data pipelines that offer different data types depending on the platform	Gathering data from websites using crawlers/spiders/scrapers.	Donation of existing digital traces with informed consent	Client-side tracking software that is installed with informed consent
Time frame of collected data	Retrospective	Retrospective/Prospective	Retrospective (collects existing digital trace data)	Prospective (tracks digital traces as they are produced)
Types of data	Includes published and public data from digital platforms	Data available on the websites	Includes non- or semi-public user data and data not visible to user (e.g., profiling, etc.)	Includes (mostly nonpublic) behavioral sequence data (e.g., click streams, screenshots, etc.)
Robots.txt	Yes (for private APIs)	Yes	n/a	n/a
Terms of Service	Yes	Sometimes (accounts/popups)	No	No
GDPR	When personal data is involved	When personal data is involved	When personal data is involved	When personal data is involved
Consent of participants	No	No	Yes	Yes
Potential for reactivity/ social desirability biases	Low	Low	Low	Medium to high
Level of gathered content	(Mostly) Aggregate-level data	Individual-level data	Individual-level data	Individual-level data
Privacy risks in the collection of personally identifiable information	Medium	High	High	Very high
Main advantages	Easy and legal	Flexible	Private data; informed consent	Private data and algorithmic data; informed consent
Main challenges	Public data only; few social media APIs left or expensive	Public data only; legally complex	Expensive	Expensive and reactive

Ohme et al, 2024, Communication Methods and Measures

TODAY

Lecture

Explain what APIs and web scraping are (in your own words).

Understand how HTML code is structured

Distinguish between the role of robots.txt, Terms of Service and GDPR.

Understand the main advantages, challenges and legal considerations of APIs and Web Scraping.

Lab

Learn how to read robots.txt

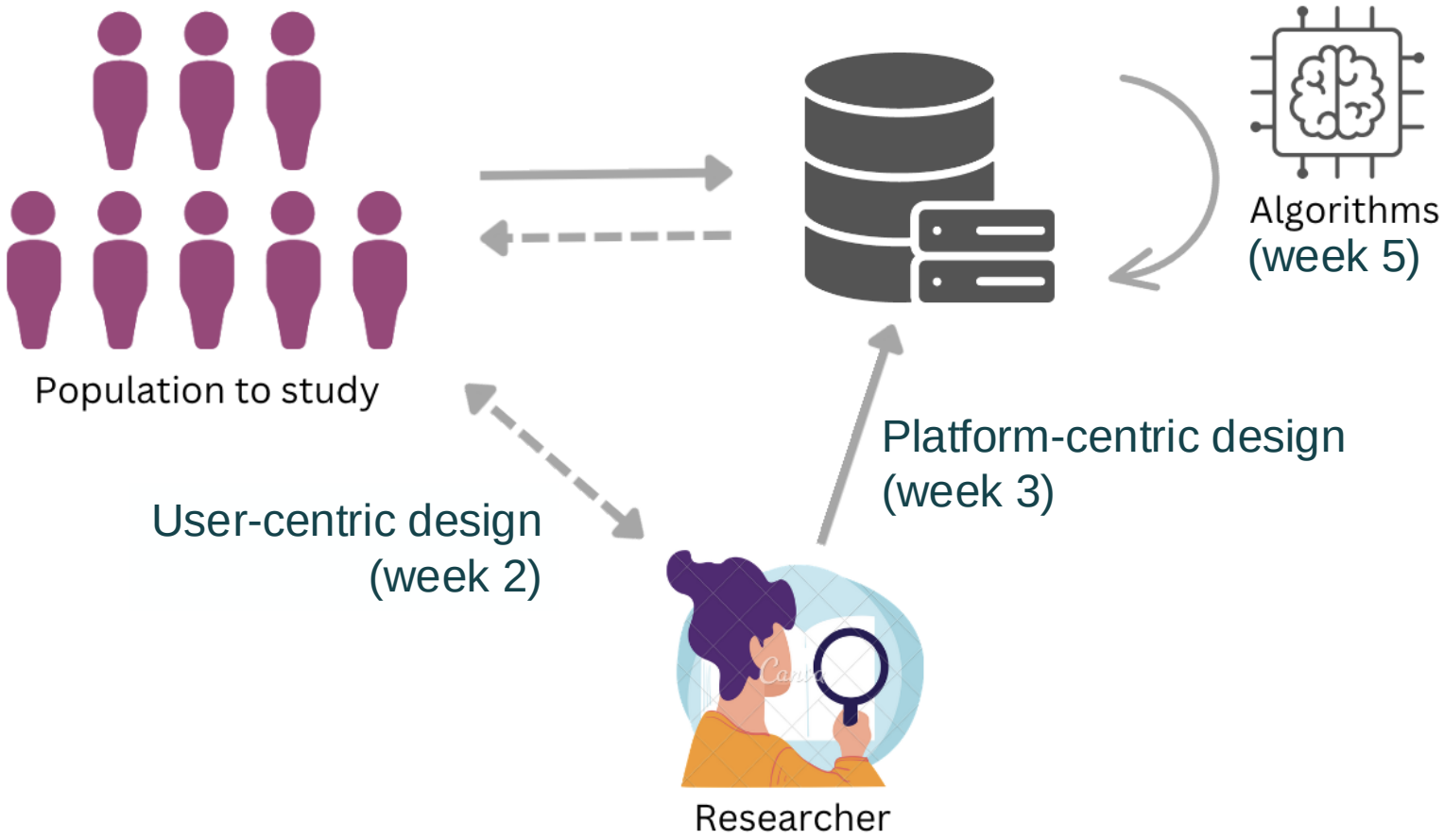
Use APIs to extract data:

- Wikimedia
- TheGuardian

Web scraping data from NU.nl

Saving text data to your hard drive

Summary of the course



Week 4: Errors in DTD
Week 6: Ethics and Legislation
Week 7: Designed big data
Week 8: Beyond DTD and Q&A

Lab today: RUPPERT - 011

Next Wednesday: second feedback moment

- Finish that you have finished gathering your data (or at least most of it)
- Start discussing any issues related to **representation** and **measurement errors** in your data. To prepare for this, please read chapters 3.1–3.4 from the book *Bit by Bit*.

Please print or write down the list of tasks that you have completed this week.

Some slides helps structure the discussion:

- Quick reminder of your RQ
- Explain what data you have gathered so far.
- Assess whether the data you collected accurately represents your target population. What gaps/biases do you expect?
- Assess whether the data allows you to measure the key concepts outlined in your RQ and what biases you expect.