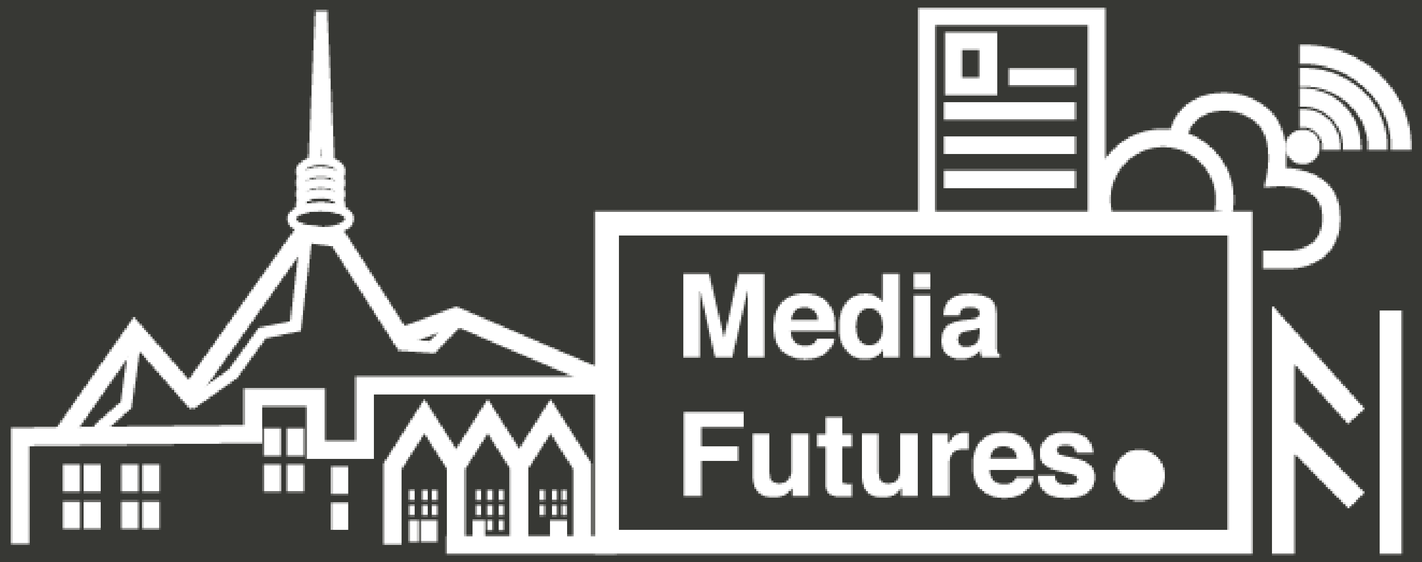



# Introduksjon til Twitter API v2 and Tweepy

INFO319 Guest Lecture 15.09.2022

sfi Research Centre for Responsible Media Technology and Innovation
Project number 309339

# Meg

- Daniel Rosnes
- 2. års masterstudent
- Ansatt hos MediaFutures

# Agenda

- Twitter API 2.0
- Tweepy
- MediaFutures twitter prosjekter
  - TweetSearcher
  - livetweets

# Twitter API 2.0

- Lansert på slutten av 2021
- Tillater automasjon av de fleste av twitters funksjoner
  - Innsamling av tweets
    - Søk
    - Strøm
  - La andre brukere logge seg inn:
    - Tweete og interagere med tweets og brukere
- Enkelte funksjoner er ikke tillatt å *automatisere*
  - Krever aktiv handling av bruker i et grensesnitt
    - F.eks. liking av tweet.

# Tilgangsnivåer

## Essential

With Essential access, you can now get access to Twitter API v2 quickly and for free!

- Retrieve 500,000 Tweets per month
- 1 Project per account
- 1 App environment per Project
- Limited access to standard v1.1 (**only media endpoints**)
- No access to premium v1.1, or enterprise

## Elevated

With Elevated access, you can get free, additional access to endpoints and data, as well as additional App environments.

- Retrieve 2 million Tweets per month
- 1 Project per account
- 3 App environments per Project
- Access to standard v1.1, premium v1.1, and enterprise

## Academic Research

If you qualify for our Academic Research access level, you can get access to even more data and advanced search endpoints.

- Retrieve 10 million Tweets per month
- Access to full-archive search and full-archive Tweet counts
- Access to advanced search operators

# Hvordan få tilgang

- Essential: (Eksperimentering)
  - Trenger bare søke
    - Men mister det ved brudd på vilkårene!
- Elevated: (Utvikling)
  - Krever et tydelig formulert prosjekt/formål
    - Må definere:
      - Hvilke deler av APIen som skal brukes
      - Hvordan disse delene skal brukes
- Academic Research: (Forskning)
  - Krever et klart definert *forskningssprosjekt*
    - F.eks. prosjektbeskrivelse til en masteroppgave, ph.d. o.l.
    - Som Elevated, men mye mer detaljert.

# Primære forskjeller

- Må ha Academic Research tilgang for å kunne søke i hele arkivet
  - Essential og Elevated kan kun søke «recent tweets»
- Må ha Elevated (eller høyere) for å gjøre geografiske søk
- Rate limits
- Academic er også begrenset til ikke-kommersiell bruk

https://developer.twitter.com/en/docs/twitter-api/getting-started/about-twitter-api#v2-access-level

# Tweet cap

- Antall tweets fått via API v2 pr måned
  - Essential: 500 000, Elevated 2 000 000, Academic: 10 000 000
- «Fått via?»
  - I all hovedsak gjelder dette:
  - Search tweets
  - Filtered stream
- Teller ikke mot cap:
  - Tweets Lookup
  - Sample Stream

# Tweet cap

- Filtered stream har en cap på 50 tweets / sekund
- 180 000 tweets / time
- Månedscap på essential vil fylles på under 3 timer
  - 11 timer for Elevated, 55 timer for Academic
- Viktig å ha gode filtreringsregler ved bruk av filtered stream
  - Sample stream teller ikke mot cap, men har tilsvarende begrensning på tweets / sekund
- Search Tweets har en begrensning på 180 søk på 15 minutter
  - Hvert søk returnerer opp til 100 tweets
  - 72000 / time ~7 timer for å fylle cap

# Search Tweets

- Returnerer tweets fra siste 7 dager basert på spørring
- Default 10, max 100
  - Støtter «pagination»
- Kan spesifiseres med:
  - Starttid (innenfor 7 dager)
  - Sluttid (innenfor 7 dager)
  - Før eller etter spesifikk tweetid
  - (og mye mer via spørring)

# Filtered Stream

- Returnerer tweets idet de er postet, basert på en filtreringsregel
- Maks 50 tweets pr sekund
- Regler kan settes og fjernes mens streaming pågår
  - 5 regler på essential
  - 25 på elevated
  - 1000 på academic

# Fields

- De returnerte objektene har bare et par datapunkter
  - Tweets har tekst og id
- For å få mer data må vi be om det
  - «Expansions»
  - «tweet_fields» - data som omhandler tweeten
  - «user_fields»- data som omhandler bruker
  - «media_fields»- data som omhandler media
  - «poll_fields» - data som omhandler meningsmåling
  - «place_fields» - data som omhandler plasseringsdata (kun academic)

# Tweet objektet

- Returnerer id og tekst om ikke annet er spesifisert
- Nevneverdige tillegg:
  - author_id
  - created_at
  - attachments
  - public_metrics

  - entities
  - context_annotations

https://developer.twitter.com/en/docs/twitter-api/data-dictionary/object-model/tweet

# Media objektet

- Returnerer id og type (bilde/video o.l.) om ikke annet er spesifisert
- Nevneverdige tillegg
  - url
  - preview_image_url
  - height
  - width
  - alt_text
  - public_metrics (view count)

https://developer.twitter.com/en/docs/twitter-api/data-dictionary/object-model/media

# User objektet

- Inneholder id, navn og brukernavn om ikke annet er spesifisert
- Nevneverdige tillegg
  - description (bio)
  - profile_image_url
  - public_metrics
  - verified
  - (andre data som url, location basert på data brukeren har oppgitt)
- Vær obs på GDPR

https://developer.twitter.com/en/docs/twitter-api/data-dictionary/object-model/user

# Queries og Rules

- Utvidede søk med en rekke operatorer
- Mindre forskjeller mellom hvilke operatorer man har tilgjengelig
  - Begge typer bygger på logiske operatorer som «og», «eller», negasjon og union.

- https://developer.twitter.com/en/docs/twitter-api/tweets/search/integrate/build-a-query

- https://developer.twitter.com/en/docs/twitter-api/tweets/filtered-stream/integrate/build-a-rule

- https://github.com/twitterdev/getting-started-with-the-twitter-api-v2-for-academic-research/blob/main/modules/5-how-to-write-search-queries.md

# Entities og Context Annotations

- Entities er spesifikke objekter i teksten:
  - Hashtags
  - Mentions
  - Urler
  - O.I.
- Inkluderer også Context Annotations
  - Nytt for API 2.0
  - domene.entitet – f.eks. Person: Jonas Gahr Støre (og Politician: Jonas Gahr Støre)
  - Annoteringer knyttet opp mot en taksonomi

# Tweepy

- Et python bibliotek laget for interagering med Twitters API
- Oversetter «API-språk» til python kode
- Håndterer også en del feilmeldinger fra Twitter
  - Bl.a. reconnecter den streams
  - Kan settes til å vente på rate-limits

# Hovedkompoenter

- Client – klasse som håndterer enkeltforespørsler
  - Response – objektet disse returnerer
- StreamingClient – klasse som håndterer streams
  - StreamRespons – objektet som stream returnerer
  - StreamRule – objekt som «holder» på filtreringstregler

# Client

- Har metoder for de fleste av API endpointene
- Kan deles opp i 5 kategorier basert på Twitter produkter:
  - Tweets
  - Users
  - Spaces
  - Lists
  - Compliance – spesielt endpoint for datahåndtering

https://docs.tweepy.org/en/stable/client.html#

# Pagination

- Mange av Client's metoder støtter Pagination
- Paginator er en egen klasse i Tweepy
- Håndterer Pagination i Twitters API

- Eksempel:
  - tweepy.Paginator(tweepy.Client.search_recent_tweets, query='INFO319').flatten(limit=1000)
  - Vil returnere 1000 resultater istedenfor 10(default) 100(max).

https://docs.tweepy.org/en/stable/v2_pagination.html

# StreamingClient

- Klasse som håndterer strømmefunksjonene i APIet
- To «primære» metoder
  - .sample
  - .filter
- Regel metoder
  - add_rules
  - delete_rules
  - get_rules

https://docs.tweepy.org/en/stable/streamingclient.html

# StreamingClient

- Har også metoder for håndtering av responser
  - on_data – mottar de rå json dataene fra APIet
  - on_response – mottar samme data pakket inn i et StreamResponse objekt
    - Underkategorier av response med kun individuelle responser:
      - on_tweet
      - on_includes
      - on_errors
      - on_matching_rules – APIet returnerer tagen på regelen som har matchet en tweet – kan brukes til logisk seperasjon av filtrene
- Har også typiske socket connection metoder som on_connect, on_closed, on_disconnect o.l.

# MediaFutures prosjekter med Twitter API

- TweetSearcher
- livetweets

# TweetSearcher

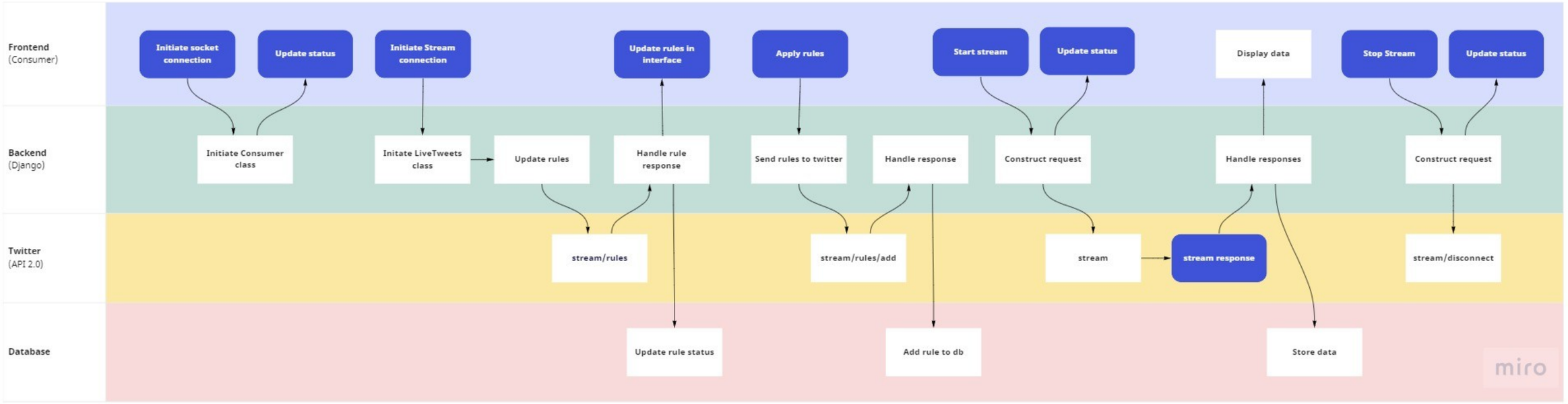
- Et verktøy for å finne kontekstuell info om en tweet
- API ble brukt til å hente ut bilde og tekst
  - Også brukt til å generere eksempel-tweets
- Bildesøk består av et enkelt reverse image søk på google
- Tekstsøk:
  - Henter ut en setning fra teksten
  - Bruker den til å søke på google news
  - Gjør et TD-IDF søk på et ett med nyhetsartikler
  - Rangerer setninger i nyhetsartiklene basert på match opp mot setningen fra tweet

# livetweets

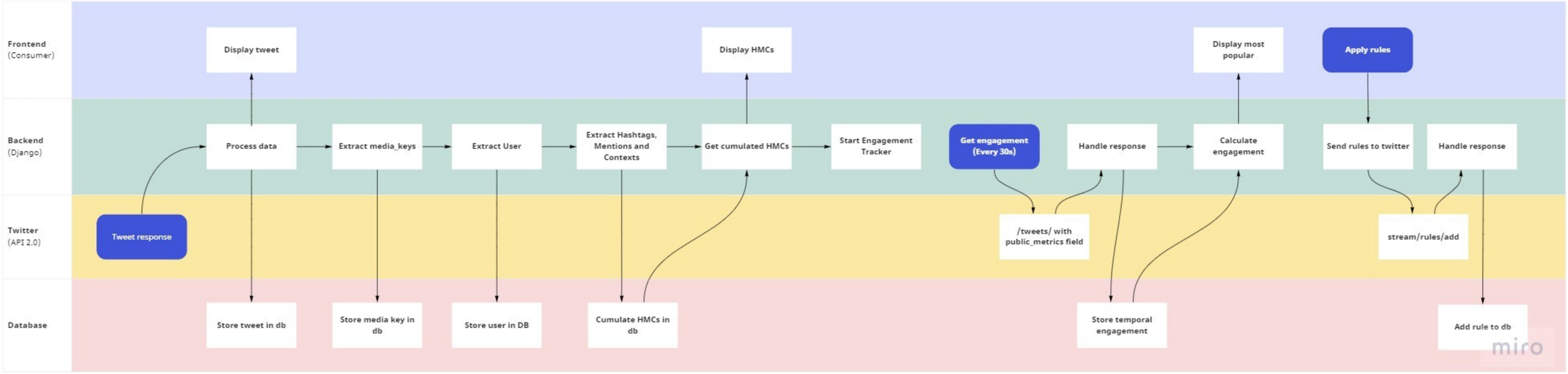
- Pågående prosjekt
- MediaFutures ble kontaktet av VG-Live
  - Ønsker å «piffe opp» VG-Live dekning av fotballkamper med SoMe-poster
- Ide: Bruke Twitter API til å fortløpende finne relevante Tweets

- Bygget på Django
- Bruker AsyncStreamingClient som base,
- Sender data over Websocket (django-channels)

# Overview of main working of the app



# Diagram of the app while running stream



Media Futures

# Thank you

## for your attention


Contact information:

Daniel Rosnes daniel.rosnes@uib.no



sfi Research Centre for Responsible Media Technology and Innovation

Project number 309339