Proposal CAB432 assignment-2

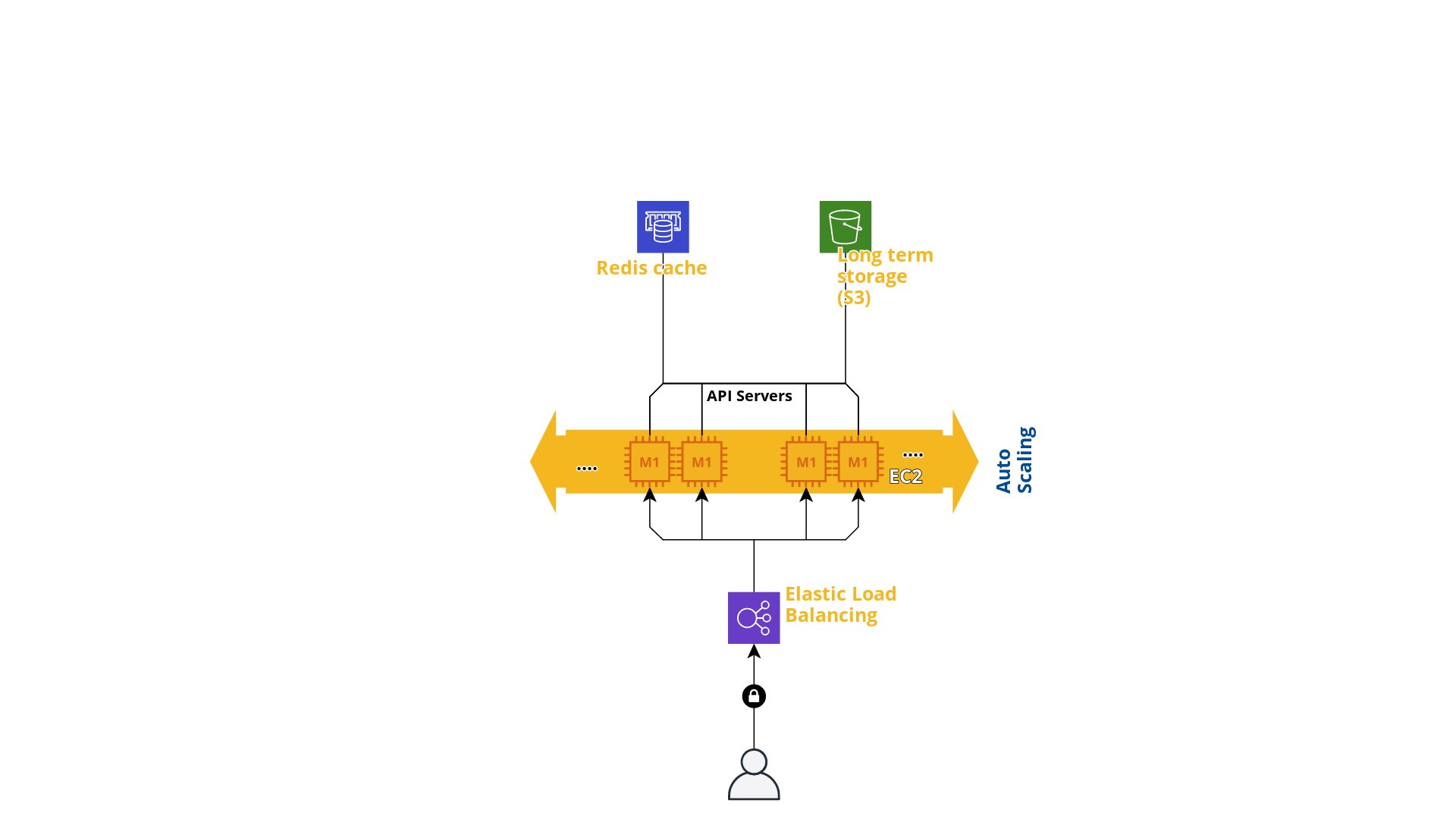
# Application purpose and description

The application takes in user queries (specified hashtags) and delivers a stream of live tweets based upon those hashtags (other filters such as location and language are also supported). In addition to this the application asynchronously processes the information and produces text mining and analysis on the live tweets. Word clouds, most frequent words, most used hashtag etc.

# Phases of implementation

1. Front-end for basic queries.
2. Back-end for basic query functionality.
3. Back-end for sentimental analysis based upon the queries and live-tweets.
4. Query support coupled with basic sentimental analysis and word extraction.
5. Front-end visualization of tweet analysis.
6. More filter options and visualization of sentimental analysis with most frequent words/hashtags.
7. Multiple instances.
8. Multiple instances with manual scaling.
9. Multiple instances with automated scaling.

* S3 for the long-term storage
* ElastiCache (redis) for cache storage
* EC2 M1 machines for the instances
* ELS (Elastic load-balancing, internet-facing) for the load balancing



# Use cases

* I want to see what people’s tweets are with these specific hashtags
* I want to see what people’s tweets are with this specific language
* I want to see what the most frequent words are with these specific filters
* I want to see what the context of the tweets are with these specific filters
* I want to see which hashtag is the most popular one based on these filters
* I want to see which words are most associated with these specific hashtags.

# Services and APIs

* Twitter API (either v1.1 or Filtered Streams Lab API)
  + Twitter has several API
* NLTK for sentimental analysis
* D3js for visualization
* Twitter Bootstrap
* Natural for natural language extraction
* NLP for natural language extraction and analysis
* ElastiCache
* S3 long-term storage
* Node.js and Javascript

# The division between client-side and server-side