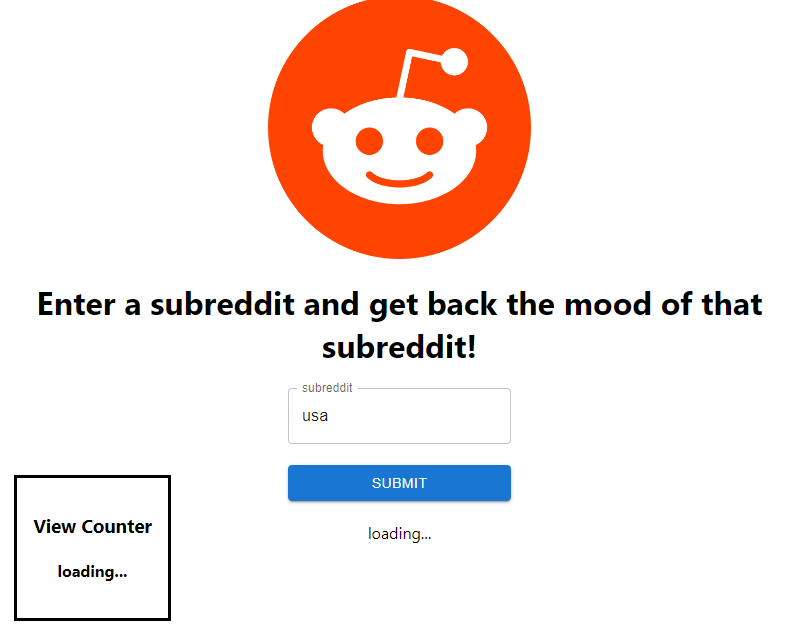
2022

<student name/s>

<student number/s>

8/12/2022

Reddit mood analyzer



Replace image with one with some relevance to your application here

CAB432 Assignment 1



Contents

[Introduction 2](#_Toc111205932)

[Mashup Purpose & description 2](#_Toc111205933)

[Services used 2](#_Toc111205934)

[Twitter Standard Search API (v.1.1) 2](#_Toc111205935)

[API 2 2](#_Toc111205936)

[API n 2](#_Toc111205937)

[Persistence Service 2](#_Toc111205938)

[Mashup Use Cases and Services 2](#_Toc111205939)

[Restaurant Search in a Foreign City 3](#_Toc111205940)

[US 2 3](#_Toc111205941)

[US n 3](#_Toc111205942)

[Technical breakdown 3](#_Toc111205943)

[Architecture and Data Flow 3](#_Toc111205944)

[Deployment and the Use of Docker 4](#_Toc111205945)

[Test plan 4](#_Toc111205946)

[Difficulties / Exclusions / unresolved & persistent errors 5](#_Toc111205947)

[Extensions (Optional) 5](#_Toc111205948)

[User guide 5](#_Toc111205949)

[Analysis 5](#_Toc111205950)

[Question 1: Vendor Lock-in 5](#_Toc111205951)

[Question 2: Container Deployment 6](#_Toc111205952)

[References 6](#_Toc111205953)

[Appendices 6](#_Toc111205954)

## Introduction

### Mashup Purpose & description

Reddit mood analyzer uses open ai to make a judgement on the overall mood of a particular subreddit based on the most popular 10 posts of that day.

For instance, on most days, checking the USA subreddit returns angry, despair, depressed etc. whereas the Australia subreddit returns, cheerful, cheeky, hopeful etc.

### 

### Services used

#### Reddit API - SnooWrap

The reddit API allows me to get the 10 hottest posts based on a given subreddit, SnooWrap provides a convenient wrapper to make interacting with this API in JavaScript easier

Endpoint: <https://www.reddit.com/dev/api/>

Docs: <https://www.reddit.com/dev/api/> https://not-an-aardvark.github.io/snoowrap/

#### Open AI GPT3

Using GPT-3 I pass the phrase “describe in 3 words the mood of a subreddit based on these posts made by users:” ~followed by the posts"

The API then makes a judgement call based on its natural language capabilities and returns something like

*happy, content, proud*

which is then returned to the request to be displayed to the user.

Endpoint https://api.openai.com/v1/completions

#### Persistence Service

Amazon s3 bucket stores a number, when page is loaded a use Effect calls an endpoint to download the value, displays it +1 then uploads the new updated value to s3

## Mashup Use Cases and Services

#### Restaurant Search in a Foreign City

|  |  |
| --- | --- |
| As a | Traveler |
| I want | To find the mood of a city |
| So that | I can avoid it if they are unhappy |

#### US 2

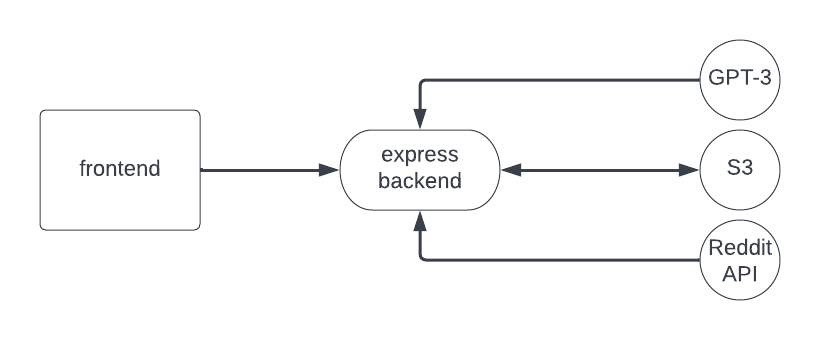
|  |  |
| --- | --- |
| As a | Reddit user |
| I want | To be able to see what subreddits are happiest |
| So that | I can participate in them |

### Technical breakdown and Architecture and Data Flow

The frontend is a react application built with CRA that makes first a call to the backend to get the counter from AWS on the initial page load via a use Effect hook getting then incrementing the counter.

then when the user makes a search, a request is sent to the backend, the backend asks the reddit API for the top 10 posts of the searched subreddit, then it sends a fetch to GPT-3 to ask for it’s opinion on the mood sentiment of the top 10 posts, once the backend receives this data the response is sent containing the output of the open ai query.

The biggest issue in development was that the AWS keys kept changing automatically every hour, but luckily this was fixed



### Deployment and the Use of Docker

The frontend and backend are built then pushed into docker hub, a docker compose file downloads these images then deploys them in the environment

### Test plan

|  |  |  |  |
| --- | --- | --- | --- |
| Task | Expected outcome | Result | Screenshot appendix |
| Valid subreddit retrieves successfully | Result is displayed | Subreddit is retrieved successfully | 1 |
| Invalid subreddit displays error | Error is displayed in message field | “subreddit not found”  Displayed in error field | 2 |
| Subreddit with no posts | Error message, no posts | “subreddit not found”  Displayed in error field, because the check for invalid sub consists of checking if the returned posts are empty | 3 |
| Submit an empty field | Error, please enter query | Error, please enter query | 3 |

Difficulties / Exclusions / unresolved & persistent errors /

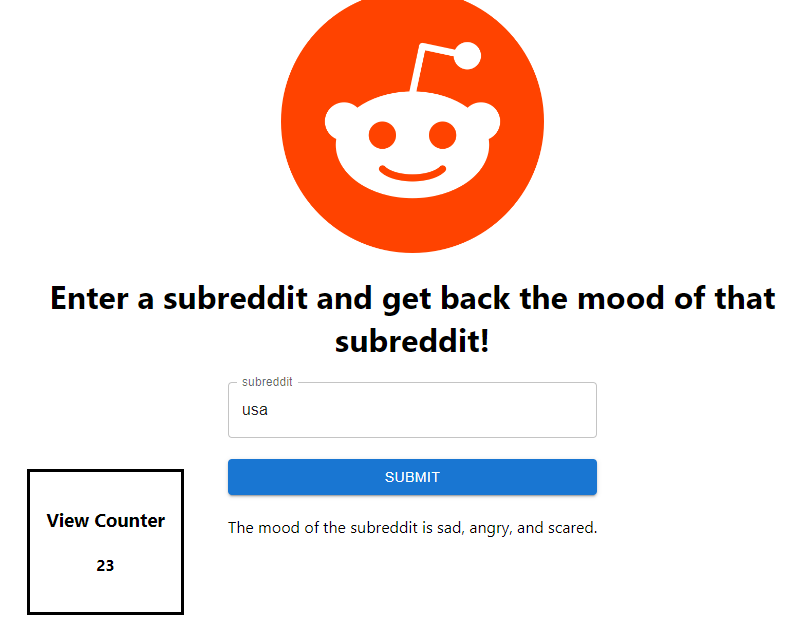
AWS keys being reset every hour was a massive hassle, I also had trouble getting the docker compose to work on the virtual machine as I had a misconception that I had to plug the build file into the docker compose somehow, but I learned that I had to plug an image into the docker compose which solved my issue

## Extensions (Optional)

For the next version I would let the user search for mood on a particular date, would be interested to see how historical events impacted sentiment, maybe if I could have the mood represented as a score 0-100 I could graph sentiment over a period of time.

## User guide

* *Find a subreddit*
* *Enter the subreddit into the search field*
* *Press submit*
* *Read output*



## Analysis

*In this section we ask you look at your application and to analyse it in response to a couple of prompts we supply below. The marking is based on the quality of the analysis and not on the length of your response. There are two questions. In each case, there is an overall question, and then a series of bullet points that help you respond. A good answer can be no more than a couple of sentences in response to each of them. Say more if you have more to say, but don’t waffle. Say it quickly and get on with the next one. This exercise is comparatively straightforward but there is a corresponding task in Assignment 2 which will be far more difficult.*

### Question 1: Vendor Lock-in

For s3, I think the vendor dependence with regards to object storage is quite low, not only do all major cloud providers offer object storage but you can even use self-hosted object storage such as minio [MinIO | High Performance, Kubernetes Native Object Storage](https://min.io/) which even supports the AWS SDK natively meaning you don’t need to change any code

On the other hand, while many natural language generation providers exist on the market, none meet the quality of GPT-3 which is also 100% closed source, in the future this is likely to change as open source initiatives meet the demand of the market for this type of service, it is worth noting though that most of the “lower quality” providers do support the OPENAI node package so porting to a new service would be trivial if the opportunity arises

*(2 marks)*

### Question 2: Container Deployment

I think if this application were to scale, I would be interested in moving the backend to a serverless function service such as GCP cloud run as this removes complications related to scaling complexities regarding the backend

I would like to use a managed auth service like AWS Cognito or firebase/GCP Auth rather than developing a solution myself as this is prone to security fault, firebase auth also handles user accounts.

I don’t think deployment to a virtual machine provides any benefits in this situation, in fact I think it would be disadvantageous by preventing us from scaling our containers using Kubernetes or something like it to meet new demand.

I think that the s3 counter we have now is appropriate for scale as it is not critical for functionality rather being simply an attachment and it’s rather unimportant that it should be perfectly accurate.

Should we implement user accounts, I would use either an object or document storage to keep the user data and the actual sign in information would be kept in the managed auth service. All major providers offer both managed auth and document databases

## Appendices

*Screenshots*

*1*

Graphical user interface, application

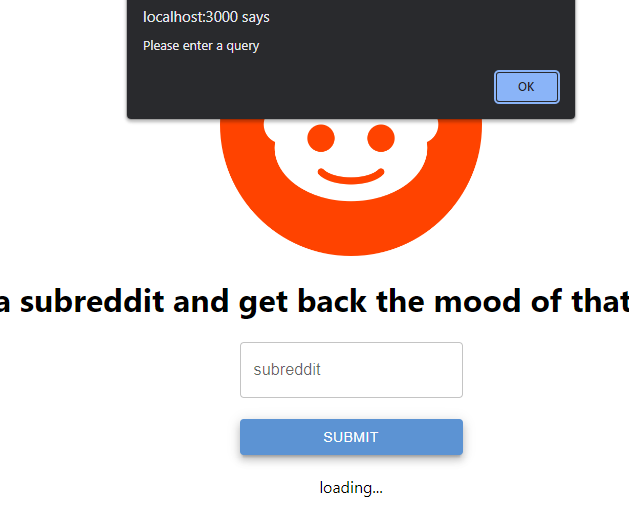
Description automatically generated

*2*

*Graphical user interface

Description automatically generated*

*3*

**