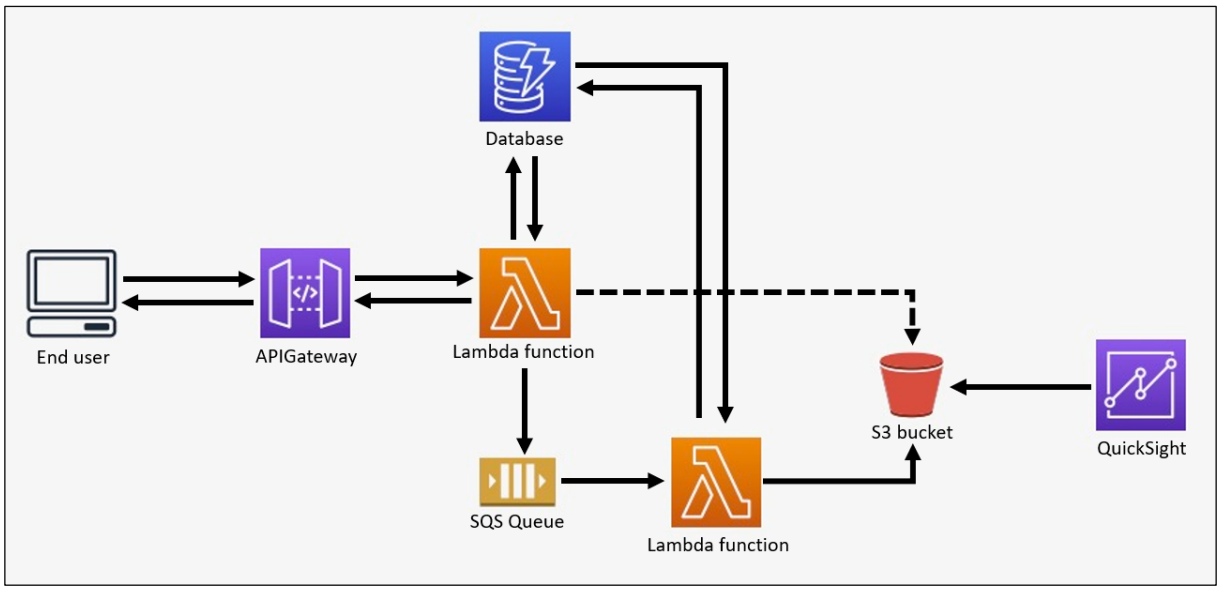
Here is my suggestion to implement a solution for logging and analyzing user searches on the Offerwall page.

Given that the website already exists, and the search bar makes a fetch request to retrieve results for display (matching brands), I recommend that Glady establishes the following infrastructure:



1. S3 Bucket and RESTful API

First, we need to create an S3 bucket and set up a RESTful API using Amazon API Gateway. This API will handle requests from the search bar and trigger a Lambda function to process the input.

Assuming user identifications are stored in the request header, the Lambda function will retrieve user details from the existing database. These details may encompass data markers such as name, age, gender, and other details like location (city) or the device used to make the request.

This data will be transformed into a JSON file, which will be uploaded to the previously created S3 bucket. If this process impacts user experience, we can implement an SQS queue. The Lambda function can publish only essential details from the header and the search query to this queue. The queue can be configured to invoke another Lambda function, which, in turn, will retrieve user details, construct the JSON, and store the data in the S3 bucket.

The Lambda function should also execute brand searches and generate matching results.

2. Amazon QuickSight Integration

An Amazon QuickSight account should be created using our existing Amazon account. Upon account creation, make a manifest.json file with the following content:

```
{
    "fileLocations": [
        {
            "URIs": [
                "s3://my-bucket/"
            ]
        }
    ],
    "globalUploadSettings": {
        "format": "JSON"
    }
}
```

The manifest.json should then be uploaded to QuickSight by navigating to DataSets and selecting the S3 bucket as our data source.

3. Data Visualization

With the datasets in place, we can start visualizing the data on our dashboard.

The choice of using S3 is driven by its reliability, scalability, and cost-effectiveness as an object storage service capable of storing and safeguarding extensive data volumes. Additionally, S3 allows the decoupling of compute and storage, enabling each to be scaled independently. S3 seamlessly integrates with a variety of AWS technologies, including QuickSight, Athena, and OpenSearch.

Amazon QuickSight was my preferred choice over Athena because QuickSight has a powerful and intuitive user interface that allows users create dashboards and interactive visualizations with low to no code options. It also has a unique feature called ML Insights, which uses machine learning to automatically discover hidden patterns, outliers, anomalies, and forecasts from the data. Athena on the other hand is mainly a query service that does not have the same features and capabilities as QuickSight. Athena can be used to prepare and transform data for QuickSight, but it cannot provide the same level of data visualization and analysis as QuickSight.