PARSHVANATH CHARITABLE TRUST'S

A.P. Shah Institute of Technology

Thane, 400615

Academic Year: 2023 - 2024

Department of Computer Engineering

CSL605 SKILL BASED LAB COURSE: CLOUD COMPUTING

Mini Project Report

- **Title of Project** : Interactive Dashboard
- **Year and Semester** : T.E. (Sem VI)
- **Group Members Name and Roll No.** : Tejas Bhandary (14)
  Aaryan Chothani (20)
  Atharva Dalvi (21)

Table of Contents

Problem Definition:

In the contemporary landscape of software companies, the volume and complexity of data are reaching unprecedented levels. This influx of data, comprising various metrics, user feedback, operational statistics, and more, poses significant challenges in terms of efficient management, analysis, and utilization. Conventional data tracking tools often fall short in accommodating the vast and dynamic nature of this data, making it challenging for companies to derive meaningful insights and make informed decisions in real-time.

To address this pressing issue, there arises a need for a comprehensive solution that not only facilitates the efficient management of vast datasets but also provides intuitive visualization and analysis capabilities. An interactive dashboard emerges as a viable approach to tackle these challenges effectively. By consolidating diverse data streams into a centralized platform, a dashboard offers a holistic view of the company's operations, performance metrics, and key indicators.

Key objectives of the project include:

1. Data Aggregation and Integration: The dashboard will integrate with various data sources, including databases, and cloud services. This integration ensures seamless aggregation of data from disparate sources, providing a unified view for analysis.

2. Real-time Data Updates: Given the dynamic nature of data in software companies, real-time updates are imperative for accurate decision-making. The dashboard will support real-time data updates, enabling users to access the latest information instantaneously.

3. Intuitive Visualization: Complex datasets often require intuitive visualization techniques to derive actionable insights. The dashboard will employ interactive charts, graphs, and visualizations to represent data trends, patterns, and anomalies effectively.

4. Customization and User Accessibility: Recognizing the diverse needs of users within the organization, the dashboard will offer customization options to tailor views and metrics according to individual preferences. Moreover, it will ensure accessibility across devices and platforms, enabling seamless access for stakeholders.

By addressing these objectives, the interactive dashboard will empower software companies to harness the full potential of their data assets, facilitating informed decision-making, performance monitoring, and strategic planning.

Introduction:

In the rapidly evolving landscape of technology and software development, data reigns supreme as the cornerstone of success. Software companies, ranging from startups to industry giants, are inundated with vast amounts of data generated from various sources such as user interactions, system logs, financial transactions, and performance metrics. This deluge of data presents both opportunities and challenges, as companies strive to harness its potential to drive innovation, improve operational efficiency, and gain a competitive edge in the market.

However, the sheer volume and complexity of data pose significant hurdles in its effective management, analysis, and interpretation. Traditional data tracking tools and spreadsheets, while useful for basic data management tasks, fall short when confronted with the scale and dynamism of modern data ecosystems. As a result, there arises a pressing need for advanced solutions that can aggregate, process, and visualize data in real-time, empowering companies to derive actionable insights and make informed decisions swiftly.

In response to this critical need, we introduce a groundbreaking project focused on developing an interactive dashboard for software companies, leveraging the Node.js in conjunction with AWS services. This project aims to revolutionize the way software companies manage and analyze their data, providing a comprehensive and intuitive platform for data-driven decision-making and strategic planning.

Understanding the Data Challenge:

The modern software landscape is characterized by an exponential growth in data generation, driven by factors such as increasing user engagement, the proliferation of connected devices, and the adoption of cloud-based technologies. This exponential growth presents software companies with a dual challenge: how to efficiently manage the sheer volume of data and how to extract meaningful insights from it in a timely manner.

Conventional data management approaches, reliant on manual data entry, disparate spreadsheets, and siloed databases, are no longer adequate to cope with the demands of today's data-driven world. These approaches often lead to data fragmentation, inconsistencies, and latency issues, hindering companies' ability to gain a holistic view of their operations and make informed decisions.

Moreover, the dynamic nature of software companies, characterized by rapid iterations, evolving user requirements, and fluctuating market conditions, necessitates a real-time approach to data analysis and decision-making. Static reports and batch processing

methods are ill-suited to address the need for timely insights and actionable intelligence, leaving companies vulnerable to missed opportunities and suboptimal outcomes.

Interactive Dashboard:

In light of these challenges, an interactive dashboard emerges as a promising solution to revolutionize data management and decision-making in software companies. By consolidating diverse data streams into a centralized platform and providing intuitive visualization and analysis capabilities, an interactive dashboard offers a comprehensive view of the company's operations, performance metrics, and key indicators.

At its core, the interactive dashboard serves as a dynamic interface between the company's data assets and its stakeholders, facilitating seamless data exploration, analysis, and collaboration. Through interactive charts, graphs, and visualizations, users can uncover hidden patterns, identify emerging trends, and monitor key performance indicators in real-time, empowering them to make data-driven decisions swiftly and confidently.

Furthermore, the interactive nature of the dashboard enables users to customize views, drill down into specific data subsets, and generate ad-hoc reports tailored to their unique requirements. This flexibility ensures that stakeholders across the organization, from executives to frontline employees, have access to relevant insights and actionable intelligence, fostering a culture of data-driven decision-making and continuous improvement.

In the subsequent sections, we delve deeper into the objectives, methodology, and expected outcomes of this project, outlining our approach to designing and implementing an interactive dashboard solution that addresses the evolving needs and challenges of software companies in the digital age. Through collaboration, innovation, and a relentless focus on user experience, we aim to deliver a transformative solution that unlocks the full potential of data as a strategic asset for software companies worldwide.

Description:

This project is created using Amazon Web Services (AWS) Cloud services and Node.js libraries including Express.js and React.js. The brief description about every cloud service used in this project is mentioned below:

1. AWS Relational Database Service (RDS):

Amazon Relational Database Service (Amazon RDS) is a web service that makes it easier to set up, operate, and scale a relational database in the AWS Cloud. It provides cost-efficient, resizable capacity for an industry-standard relational database and manages common database administration tasks. It is simple to set up, operate, and scale with demand. Amazon RDS automates the undifferentiated database management tasks, such as provisioning, configuring, backups, and patching.

In this project, we have harnessed the power of Amazon RDS (Relational Database Service) alongside MySQL to effectively manage our database housing crucial sales data. RDS provides a robust and managed environment for MySQL. This database facilitates the storage, retrieval, and manipulation of sales data essential for generating insights for the dashboard.

2. AWS Elastic Compute Cloud (EC2):

Amazon Elastic Compute Cloud (Amazon EC2) provides on-demand, scalable computing capacity in the Amazon Web Services (AWS) Cloud. Using Amazon EC2 reduces hardware costs so you can develop and deploy applications faster. You can use Amazon EC2 to launch as many or as few virtual servers as you need, configure security and networking, and manage storage. You can add capacity (scale up) to handle compute-heavy tasks, such as monthly or yearly processes, or spikes in website traffic. When usage decreases, you can reduce capacity (scale down) again. You can add capacity (scale up) to handle compute-heavy tasks, such as monthly or yearly processes, or spikes in website traffic. When usage decreases, you can reduce capacity (scale down) again.

We have connected out RDS database instance to an EC2 instance. This helps to manage and query the database. By connecting to the EC2 instance, we can easily query the MySQL database for any data and manipulation.

3. AWS Elastic Beanstalk:

With Elastic Beanstalk, you can quickly deploy and manage applications in the AWS Cloud without having to learn about the infrastructure that runs those applications. Elastic Beanstalk reduces management complexity without restricting choice or control. You simply upload your application, and Elastic Beanstalk automatically handles the details of capacity provisioning, load balancing, scaling, and application health monitoring.

Elastic Beanstalk supports applications developed in Go, Java, .NET, Node.js, PHP, Python, and Ruby. When you deploy your application, Elastic Beanstalk builds the selected supported platform version and provisions one or more AWS resources, such as Amazon EC2 instances, to run your application.

In this project, we have leveraged Elastic Beanstalk to deploy our Express.js backend server which queries the database and sends data to the React frontend. Elastic Beanstalk simplifies infrastructure management, allowing us to focus on application development. The Beanstalk environment was set up using Node.js as the platform.

4. AWS Simple Storage Service (S3):

Amazon Simple Storage Service (Amazon S3) is an object storage service that offers industry-leading scalability, data availability, security, and performance. Customers of all sizes and industries can use Amazon S3 to store and protect any amount of data for a range of use cases, such as data lakes, websites, mobile applications, backup and restore, archive, enterprise applications, IoT devices, and big data analytics. Amazon S3 provides management features so that you can optimize, organize, and configure access to your data to meet your specific business, organizational, and compliance requirements.

In our project, we have opted for Amazon S3 (Simple Storage Service) to host our React frontend. This choice provides us with scalable and durable object storage, ensuring reliable delivery of our static web content to users. By leveraging S3, we streamline the deployment process and ensure high availability of our frontend application, enabling seamless access for users while reducing operational overhead.

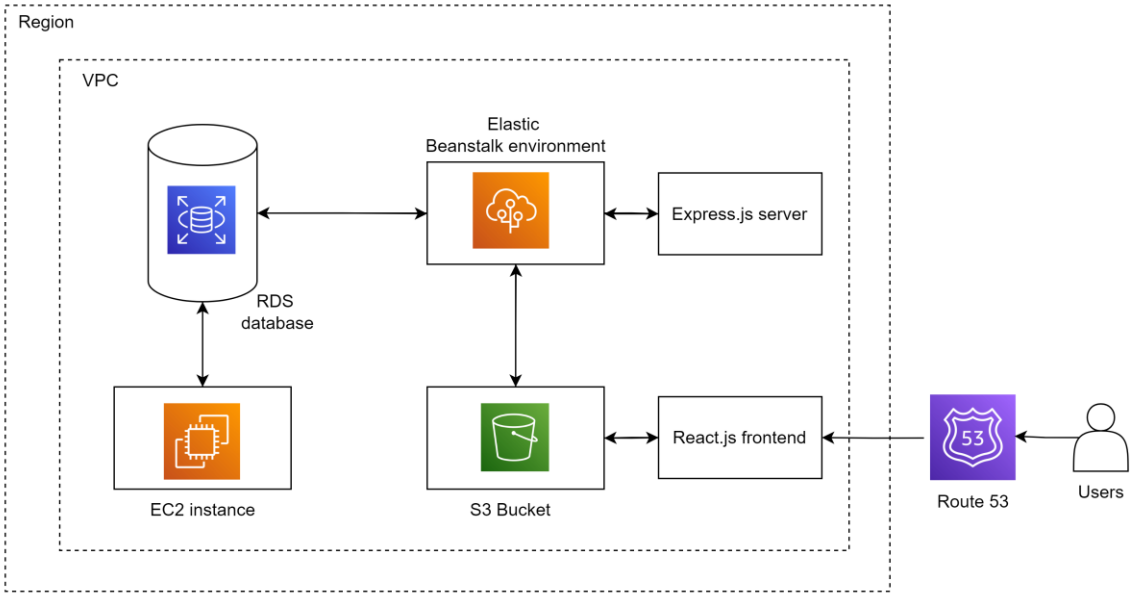

Fig1: Architecture diagram of the Project

Software Requirements:

- Express.js: The backend server is developed using Express.js, a lightweight and flexible web application framework for Node.js. Express.js simplifies route handling, middleware management, and API development. The server requires libraries like ‘mysql’ and ‘cors’ installed.

- React.js: The frontend application is built with React.js, a popular JavaScript library for building user interfaces. React.js facilitates the creation of interactive and dynamic UI components, enhancing user experience and engagement. The frontend requires libraries like ‘axios’, ‘canvasjs’, ‘toaster’ installed.

- MySQL Client (On EC2 Instance): A MySQL client is installed on the EC2 instance to facilitate database management tasks. This allows direct access to the RDS database for administration, querying, and data manipulation purposes.

Implementation details:

1. Setting Up AWS Services:
    - Set up Amazon RDS for MySQL database hosting.

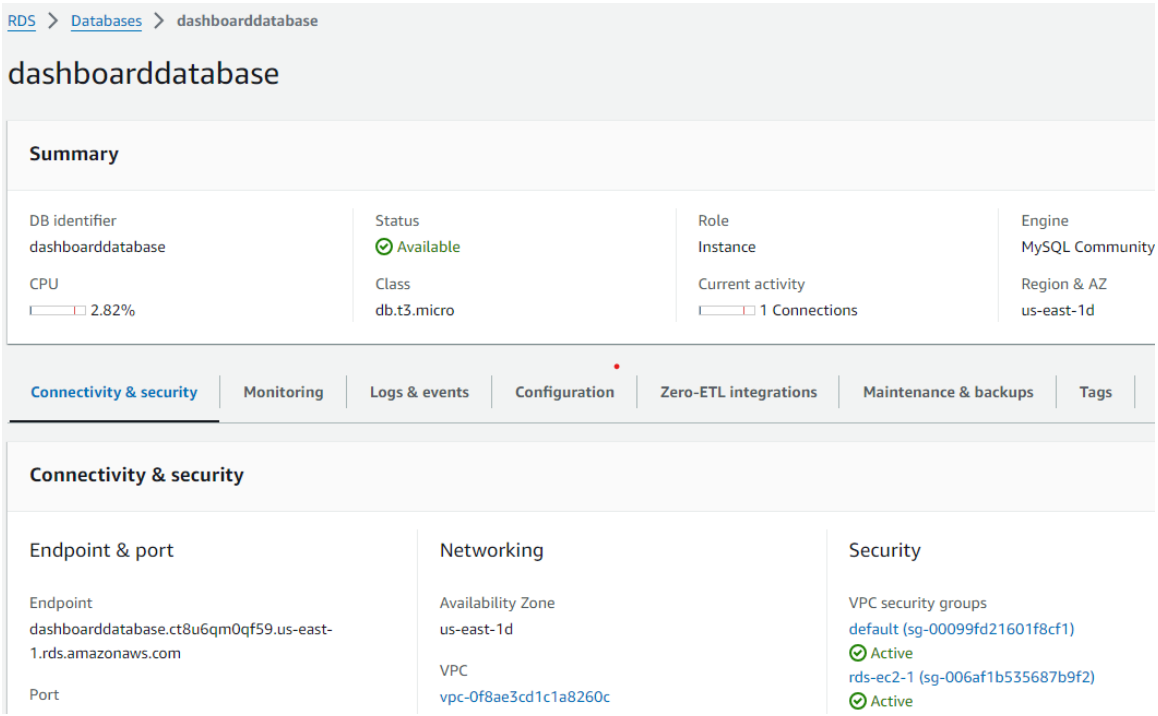

Fig 2: RDS database

- Create an Amazon S3 bucket to host the React frontend.
- Configure Elastic Beanstalk environment for hosting the Express backend.

2. Database Setup and Management:
    - Create a MySQL database schema for storing gaming sales data.
    - Import the sales dataset into the MySQL database.
    - Configure security groups and permissions for database access.
    - Connect an EC2 instance to the RDS database for management tasks.

3. Backend Development:
    - Develop the Express.js backend application.
    - Define API endpoints for data retrieval and manipulation.
    - Connect to the MySQL database using RDS credentials.

4. Frontend Development:
    - Develop the React.js frontend application.
    - Design user interfaces for data visualization and interaction.

- Implement components for displaying sales data.
- Set up routing for navigating between different views.
- Configure HTTP requests to communicate with the Express backend.

5. Integration and Deployment:
    - Integrate the frontend with the backend by specifying API endpoints.
    - Test the application locally to ensure proper functionality.
    - Package the frontend files and upload them to the S3 bucket.

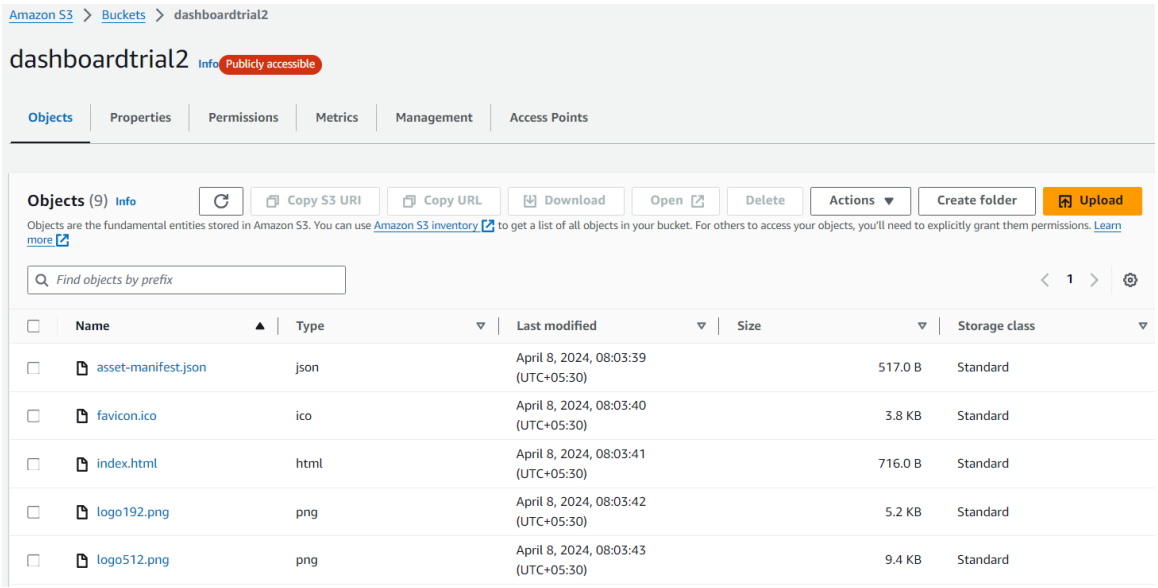

Fig 3: S3 Bucket

- Deploy the Express backend to Elastic Beanstalk.

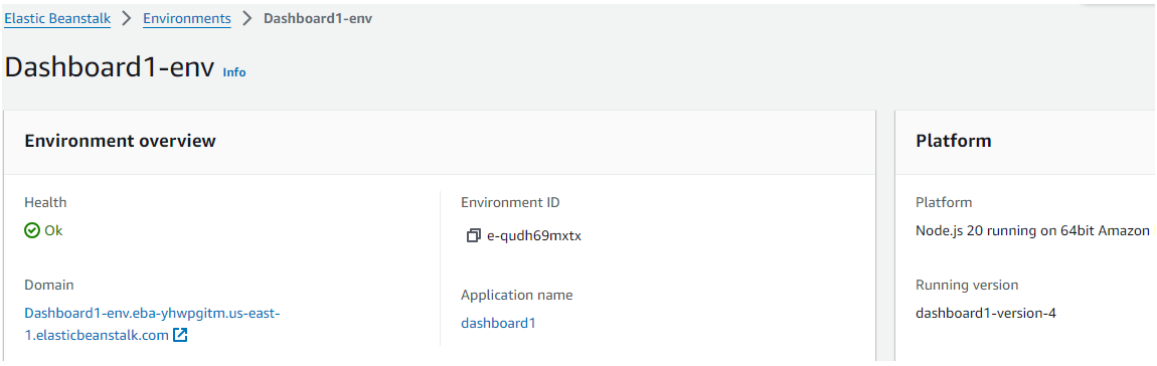

Fig 4: Elastic Beanstalk environment

- Test the deployed application to ensure it works as expected in the AWS environment.

Screenshots:

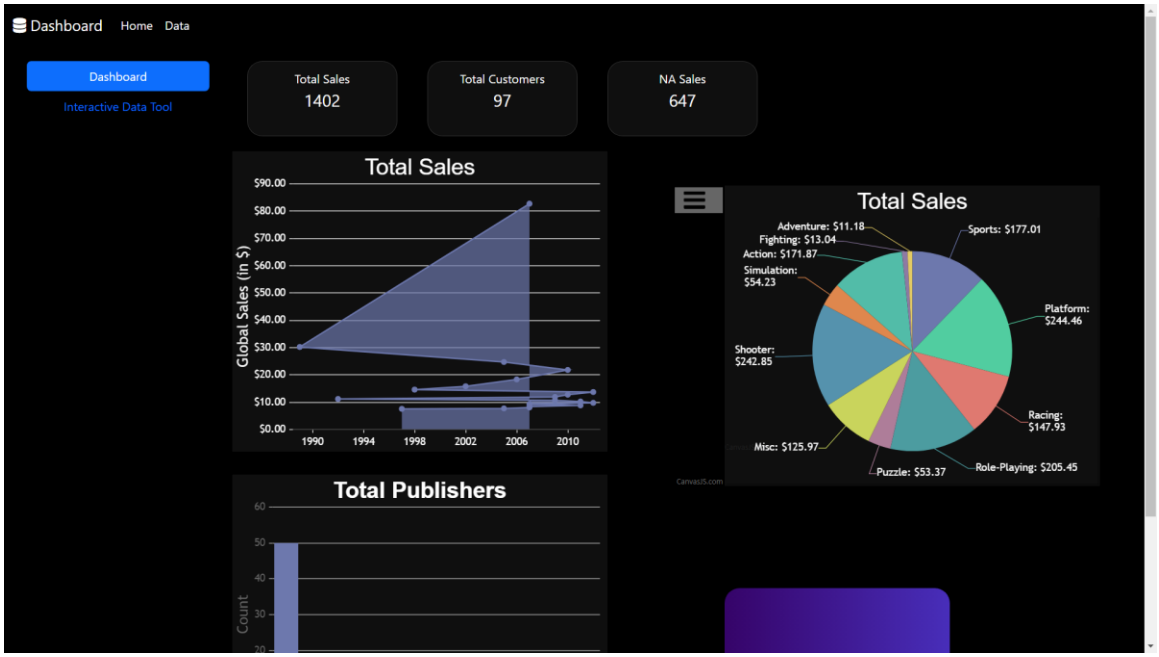

Fig 5: Main Dashboard Page - 1

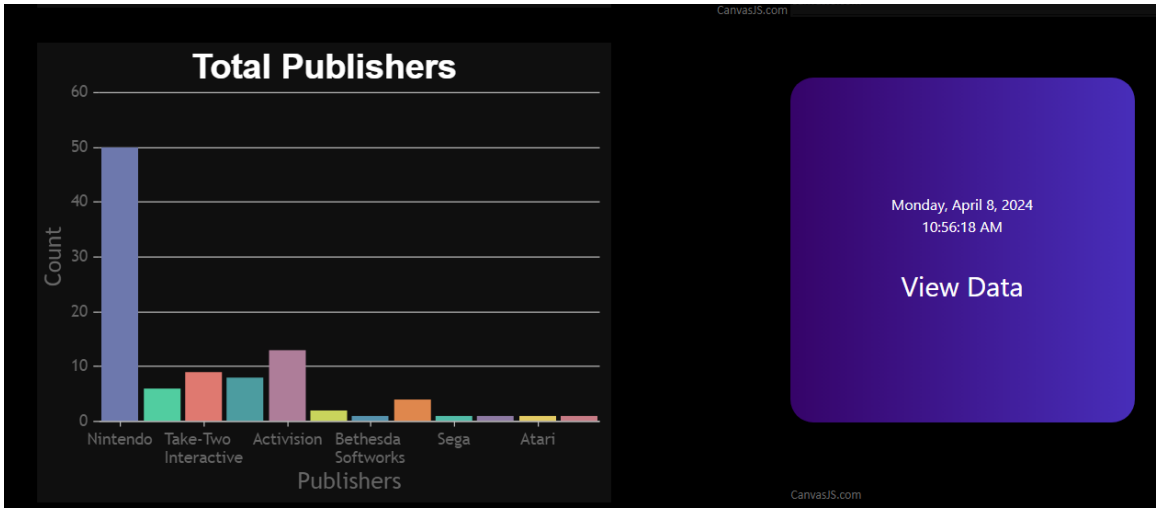

Fig 6: Main Dashboard Page - 2

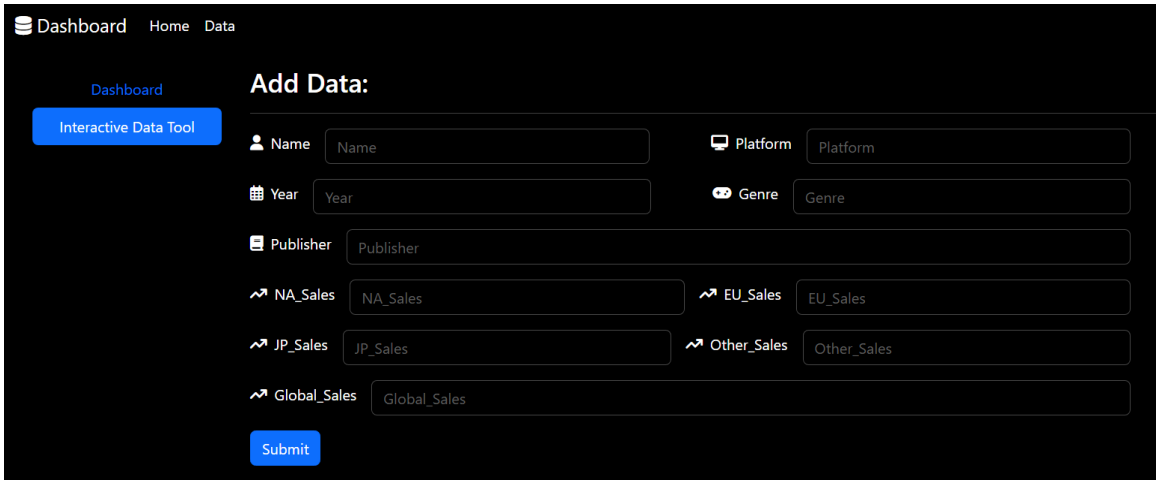

Fig 7: 'Add Data' Section to insert data

Fig 8: 'Response Data' section to delete and update data

Learning Outcome:

After the successful completion of this project, we have gained enough knowledge and learnt about Amazon Web Services (AWS) Cloud and Cloud Computing as a whole, facilitating us to create scalable, robust, easily accesible cloud applications. Through hands-on experience, we have actively built a deep understanding of frontend and backend technologies, including React.js for crafting dynamic user interfaces and Express.js for constructing robust APIs. By deploying the application on AWS Elastic Beanstalk and hosting the frontend on Amazon S3, we have gained valuable insights into cloud computing principles and deployment strategies. Additionally, actively working with Amazon RDS for MySQL database management has enhanced our skills in database design, data manipulation, and administration.

Actively taking charge of the deployment process, we have leveraged the capabilities of AWS Elastic Beanstalk to seamlessly deploy and manage the Express.js backend, ensuring scalability and reliability. Simultaneously, we have actively hosted the React.js frontend on Amazon S3, harnessing the power of cloud storage to ensure efficient delivery of web content to users across the globe. We have actively utilized Amazon RDS to host a MySQL database that serves as the backbone of the application's data storage and retrieval system. Actively designing database schemas, importing datasets, and configuring security measures, we have honed our skills in database administration and optimization, ensuring the integrity and security of your data.

By actively embracing challenges, actively seeking solutions, and actively collaborating with peers, we have actively prepared ourself for developing projects in web development and cloud computing. Armed with a diverse skill set and a proactive approach to learning, we have actively embarked on new endeavors with confidence, ready to tackle the ever-evolving landscape of technology with enthusiasm and expertise.