



DATA VISUALIZATION ADMIN DASHBOARD USING FULL STACK MERN WEB APPLICATION

¹R.Dhakshina Murthy, ²R.K.Shanmuga Priya.M.E, (Ph.D),

¹PG Scholar, ²Assistant Professor (Sr.G),

¹Department of Computer Science Engineering

¹Jai Shriram Engineering College, Tirupur, Tamilnadu, India

Abstract: Data visualization involves presenting data in graphical or pictorial form which makes the information easy to understand. It helps to explain facts and determine courses of action. It will benefit any field of study that requires innovative ways of presenting large, complex information. The advent of computer graphics has shaped modern visualization. Data visualization is the presentation of quantitative information in a graphical form. In other words, data visualizations turn large and small data-sets into visuals that are easier for the human brain to understand and process. This paper is an attempt to create Data visualizations application using MERN stack using various tools available. In the front end React js Library is used to create to render UI Components and Redux toolkit is used for state management. Along with React Js other libraries such as material ui and nivo js chart is used to render various types of visualization tools. In the back end the project is implemented using Node js & Express Js frameworks for server side operations. The project used MongoDB as a backend to store data.

IndexTerms – MongoDB, Express js, React js, Node js, MERN Stack, Data Visualization

1. INTRODUCTION

As the “age of Big Data” kicks into high gear, visualization is an increasingly key tool to make sense of the trillions of rows of data generated every day. Data visualization helps to tell stories by curating data into a form easier to understand, highlighting the trends and outliers

Data visualization is the representation of data through use of common graphics, such as charts, plots, info graphics and even animations. These visual displays of information communicate complex data relationships and data-driven insights in a way that is easy to understand.

The project was to create a data visualization dashboard using the provided JSON data. The main objective of the project was to use the MERN Stack design and develop a data visualization dashboard that would read data from a MongoDB database. The dashboard should contain different charts, graphs, and visuals that can be used to generate insights from the given data.

We are developed data visualization admin dashboard using multiple technologies. In the front end we use css, javascript, React to render the web pages. For data visualization process to create various kind of chart we used nivo.js library, mui react component library. In the backend we use node js to provide a run time environment, Express js and MongoDB which is NoSQL database to store data.

2. NEED OF THE STUDY

The objective of the project is an attempt to create Data Visualization admin dashboard using Full Stack MENR application, addressing the following challenges.

2.1. Complexity.

A highly complicated visualization could appear cluttered or make it difficult to glean valuable insights. More complexity also means users need training on the tools being used or risk creating the wrong visual type for the data being used.

2.2. Potential for misinterpretation

Users might have good intentions when using a visualization platform, but they can draw incorrect conclusions from detailed visualizations.

2.3. Data privacy and security

Users must consider the security and privacy of the data being visualized. A platform might be susceptible to cyber-attacks, thus compromising the security of data being used, or a data set could be used that isn't compliant with data privacy regulations.

2.4. Bias

Visualizations and the data they're based on should be scrutinized to ensure they aren't intentionally or unintentionally biased. Failing to do so could compromise the credibility of those analyses. For example, a data set that leaves out key demographics within a population could lead to a biased visualization of that data.

2.5. Improper visualization

The core of a lot of issues and disadvantages stems from this main one. If you're not careful in how you build your visualizations, you may end up with visualizations that don't properly convey your data. This can lead to confusion and issues down the line if you use that improper viz to do analysis and draw conclusions.

2.6. Inexact

If you're creating a visual representation of numerical data, there comes an inherent risk of creating an inexact perception of the data in the mind of the viewer. Especially if there are no keys or ways to hover over the effects to see the exact numbers involved. It'll give your viewer an idea of the data, and probably enough to draw their own conclusions, but it runs the risk of them having inexact conclusions from inexact data.

Considering all these challenges, this project aims to develop a comprehensive Data visualization Admin Dashboard Application using modern technologies and user-centric design principles. By addressing these key issues, the application intends to empower data visualization admin dashboard using modern technologies.

3. LITERATURE REVIEW

3.1. DATA VISUALIZATION

Authors: Matthew N. O. Sadiku, Adebowale E. Shadare, Sarhan M. Mus and Cajetan M. Akujobi; (2016)

Observation/Conclusion: Data Visualization involves presenting data in graphical form which make information easy to understand. It explains the summary of any vast data. It will benefit any field of study that requires innovative ways of presenting large and complex information to understand the underlying data. Advanced computer graphics has reshaped data visualization

3.2. Data Visualization Techniques for real-time information -A Custom and Dynamic Dashboard for Analyzing Surveys'

Authors: Renato Toasa, Marisa Maximiano, Catarina Reis, David Guevara; (2020)

Observation/Conclusion: Implement a generic and dynamic dashboard based on real-time information to assess the impact of the Data Visualization Techniques. The dashboard helps to interact with users based on initial set and existing set of Data Visualization Techniques

3.3. Visualization of Big Data Text Analytics in the Financial Industry

Authors: Živko Krstić, Sanja Seljan, Jovana Zoroja; (2019)

Observation/Conclusion: Textual data can decide new insights and bring precious business insights. Sources that are used for text analysis in financial industry contains internal document like emails and external documents like social media, websites, etc. Data Extraction uses to gather valuable information.

3.4. What is Interaction for Data Visualization?

Authors: Dimara, Evanthia, Perin, Charles; (2019)

Observation/Conclusion: In this paper, they define that interaction is fundamental to Data Visualization. To tackle the problems, they synthesize an inclusive view of interaction in the visualization which includes insights from information visualization, visual analytics, and scientific visualization.

3.5. DeepEye: Towards Automatic Data Visualization

Authors: Luo, Yuyu, Qin, Xuedi, Tang, Nan, Li, Guoliang; (2018)

Observation/Conclusion: Data Visualization is extremely useful to explain the significance of data to people who are visually oriented. The main task of automatic data visualization is, given a dataset to visualize its storytelling by transforming the data (e.g. selecting attributes, grouping and binning values) and deciding which visualization (e.g. bar, pie, or line charts) is the right type to a particular data. In this paper, they present DeepEye:

A novel system for automatic data visualization that tackles three problems

- Visualization Recognition
- Visualization Ranking
- Visualization Selection.

4. PROPOSED SYSTEM

The proposed system is a Data Visualization Admin Dashboard using Full Stack MERN Application tailored specifically for businesses, developed using the MERN (MongoDB, Express.js, React, Node.js) stack. This comprehensive web-based tool is designed to streamline the data visualization process.

Key features of the system include:

4.1. Admin Dashboard

The system will feature a user-friendly dashboard providing a comprehensive snapshot of homepage. It will offer insights into various options.

4.2. Geography Page

We have geography page where we can see where our users are located globally.

4.3. Sales Page Overview

We have Sales page overview where we can see number of units sold in each month. We can use this information for analyzing revenue vs. profit.

4.4. Daily Sales Chart Page

We can overview how much products sold every day by using Daily sales chart.

4.5. Monthly Sales Chart Page

Apart from daily sales chart we can also overview how much units sold on monthly basis.

4.6. Breakdown of Sales by Category

Finally using pie chart we can analyze how much products are sold in each category at any particular time.

5. RESEARCH METHODOLOGY

The development of the project application is anchored on the utilization of the MERN stack, a collection of four distinct technologies synergistically employed to create dynamic web applications and websites. The components of the MERN stack include:

5.1. MongoDB:

Serving as the database component, MongoDB is a NoSQL document-oriented database renowned for its adaptable schema and JSON-based query language. Unlike traditional relational databases, MongoDB does not enforce a predefined schema, enabling seamless storage of objects with diverse structures. This flexibility eliminates the need for early-stage schema modifications, thereby streamlining development. MongoDB's document-oriented architecture, along with its indexing capabilities, renders it a favored choice for modern businesses.

5.2. Express.js:

Collaborating with Node.js, Express.js operates as the framework layer of the MERN stack. It is an open-source server framework fully scripted in JavaScript, offering robust features for web-based and mobile application development. Express.js facilitates HTTP handling and middleware integration, furnishing developers with potent APIs and simplifying application development. Its integration with Node.js boosts development efficiency without sacrificing performance, making it a preferred framework for various JavaScript components and frameworks.

5.3. React.js:

Developed and maintained by Facebook, React.js functions as the front-end component of the MERN stack. It is an open-source client-side JavaScript library that excels at building reusable UI components, focusing primarily on the application's view layer. React.js advocates for a component-based architecture, allowing developers to craft modular and reusable components for diverse user interface elements. This approach minimizes redundant code and simplifies component logic integration within the application.

5.4. Node.js:

Serving as the runtime environment, Node.js provides an open-source, cross-platform server environment for executing scalable JavaScript applications. It facilitates the development of fast and scalable network applications, including server-side JavaScript applications. Its event-driven, non-blocking I/O architecture enhances application performance, making it suitable for highly constructing data-intensive real-time web applications. Node.js' asynchronous nature eliminates the need for waiting on API responses, thereby expediting data-intensive tasks such as audio and video file processing.

Apart from MERN Stack various frameworks are used.

5.5. nivo charts

nivo provides supercharged React components to easily build dataviz apps, it's built on top of d3. Several libraries already exist for React d3 integration, but just a few provide server side rendering ability and fully declarative charts.

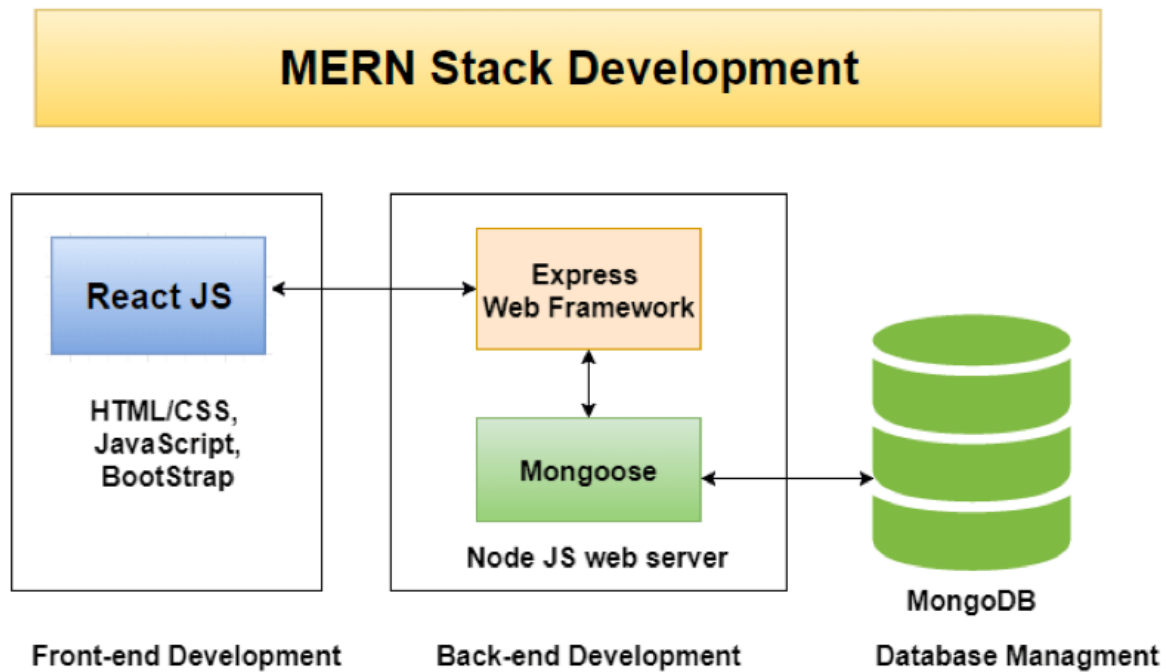


Fig 1. MERN Stack Architecture

6. IMPLEMENTATION APPROACH

The development of the Data Visualization Admin Dashboard Application using Full Stack MERN Application follows a structured implementation scheme involving several key components and steps.

This section provides an overview of the processes and technologies utilized in the development process:

6.1. Scope Definition

Define the scope of the Data Visualization Admin Dashboard Application, outlining its objectives and target users. Determine the key features and functionalities required to meet the needs effectively.

6.2. Technology Stack Selection

Choose the MERN stack as the development framework for its robustness and versatility. Select MongoDB as the database solution for its flexibility and scalability. Utilize Express.js and Node.js for server-side development to handle HTTP requests and employ React.js for building the frontend interface, ensuring dynamic and interactive user experiences.

6.3. Frontend and Backend Development

Develop the user interface utilizing React.js and nivo chart js create Dashboard Page, Sales Overview, Daily sales overview, Monthly sales overview, Breakdown of sales by category and create data visualization diagrams. Employ responsive design principles to ensure a seamless user experience across different devices. Integrate Material UI, Recharts, and Redux tools for frontend development.

For backend development, utilize Express.js and Node.js to craft server-side logic, including the creation of APIs for retrieving data. Incorporate Mongoose for MongoDB database interaction.

6.4. Database Schema Definition:

Design the database schema using MongoDB to store Sales data and define relationships between different entities to ensure efficient data retrieval and storage.

6.5. User Feedback-driven Application Improvement:

Utilize surveys, usability testing, and feedback forms to gather user feedback. Evaluate user responses to pinpoint areas requiring refinement and improvement. Continuously enhance the application progressively by addressing usability concerns and integrating new functionalities based on user input.

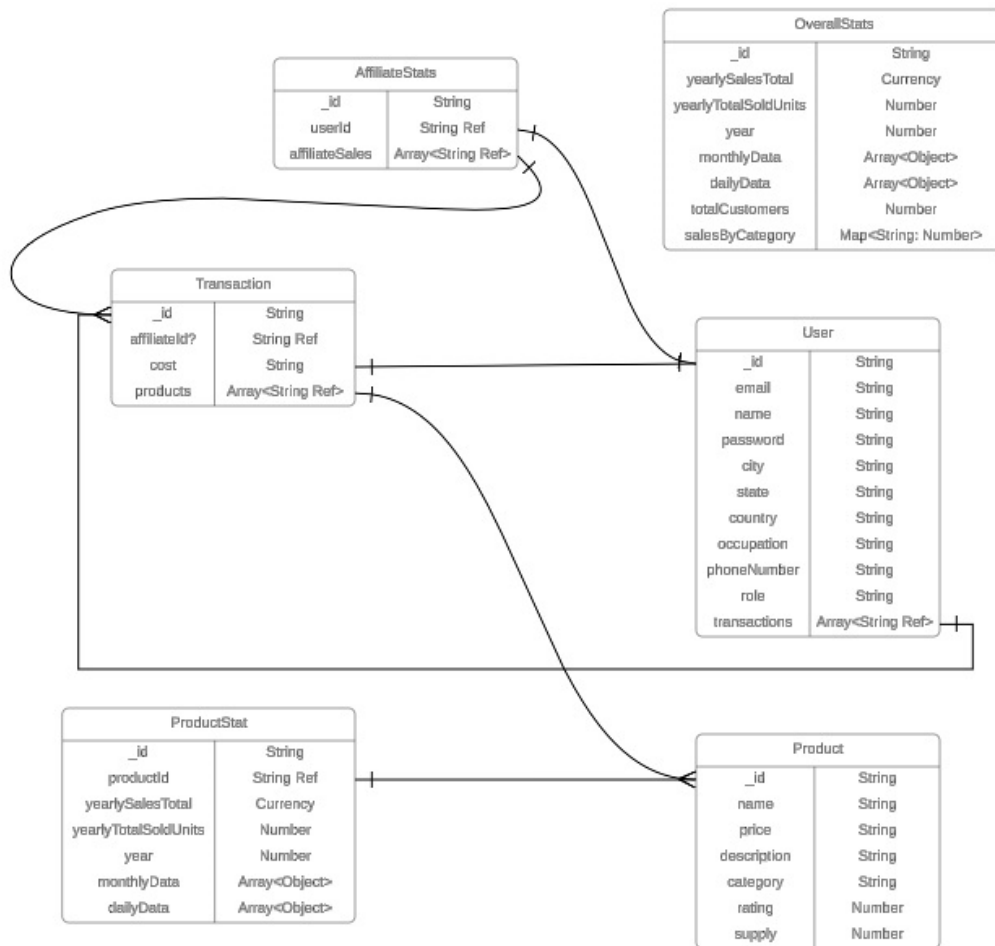


Fig 2. Data Modelling Diagram

6.6. Deployment Planning and Execution:

Prepare the application for deployment on a hosting platform or cloud server. Set up essential infrastructure, including configuring the web server, deploying the database, and managing the domain. Introduce security protocols and authentication mechanisms and monitor the application's performance.

This structured implementation approach ensures the development of a robust and user-friendly Finance Dashboard Application while adhering to best practices in software development. Careful attention is given to user feedback and iterative refinement to ensure the application meets the needs of businesses effectively.

7. RESULTS AND DISCUSSION

7.1. Admin Dashboard Page

A Dashboard page with different visualization tools such as side menu bar, line chart, Transaction table and Pie Chart. The information is coming from backend database MongoDB where the data stored in collections as json data forms.

7.2. Geography page

We have geography page where we can see where our customers are located globally. This page implemented using nivo geo chart.

7.3. Sales Page Overview

We have Sales page overview where we can see number of units sold in each month. We can use this information for analysing revenue vs. profit. This is achieved by nivo line charts.

7.4. Daily Sales Chart Page Overview

We can overview how much products sold every day by using Daily sales chart. This chart created using nivo line chart.

7.5. Monthly Sales Chart Page Overview

Apart from daily sales chart we can also overview how much units sold on monthly basis. This chart created using nivo line chart.

7.6. Breakdown of Sales by Category

Finally using nivo pie chart we can analyse how much products are sold in each category at any particular time

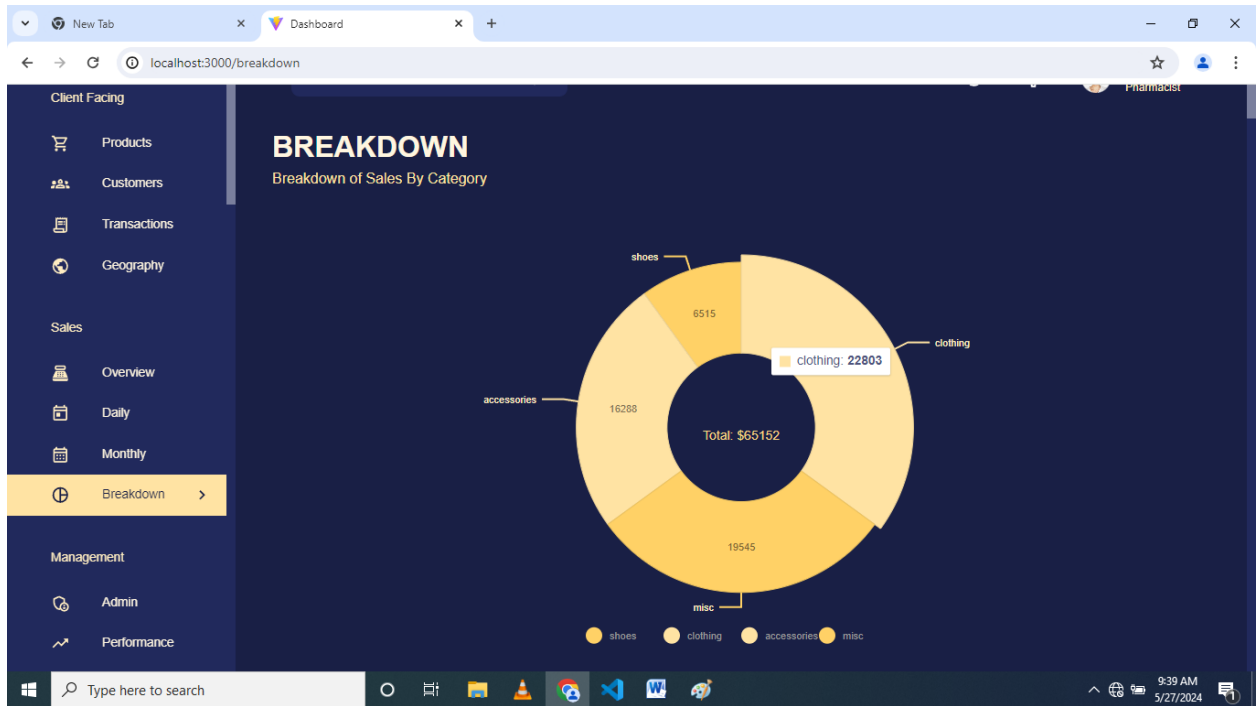


Fig 3. Breakdown of Sales by Category

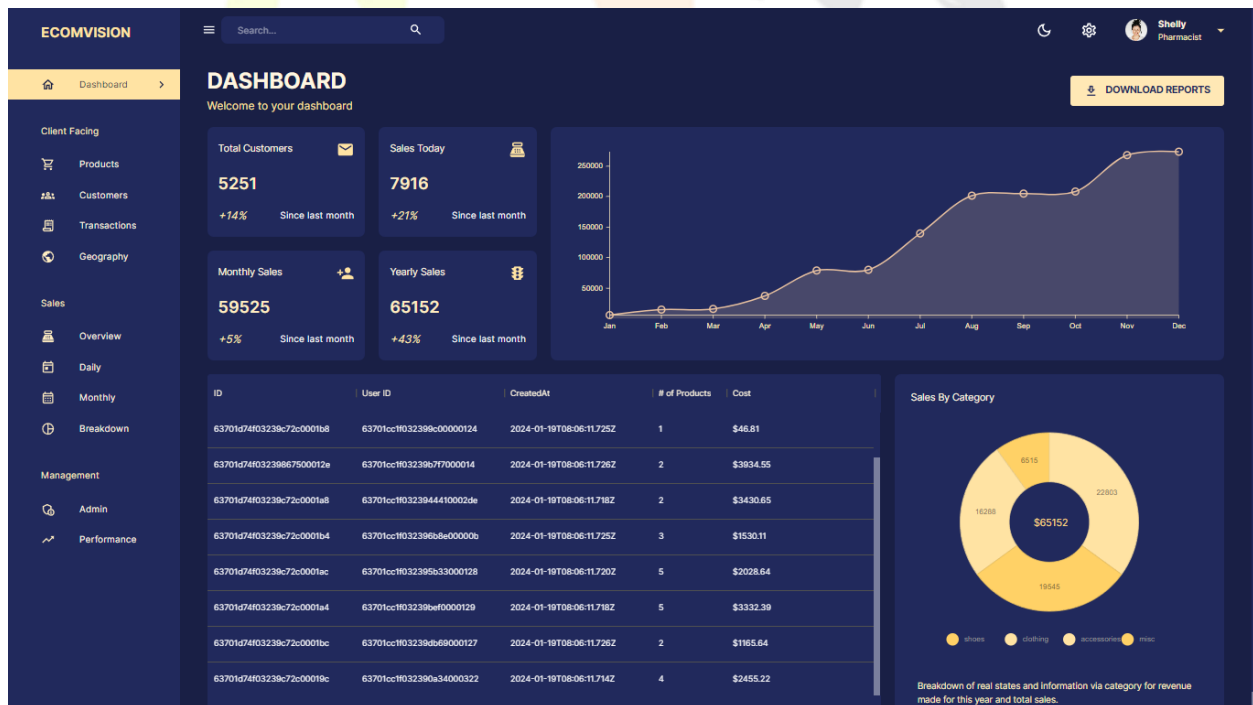


Fig.4. Admin Dashboard Page

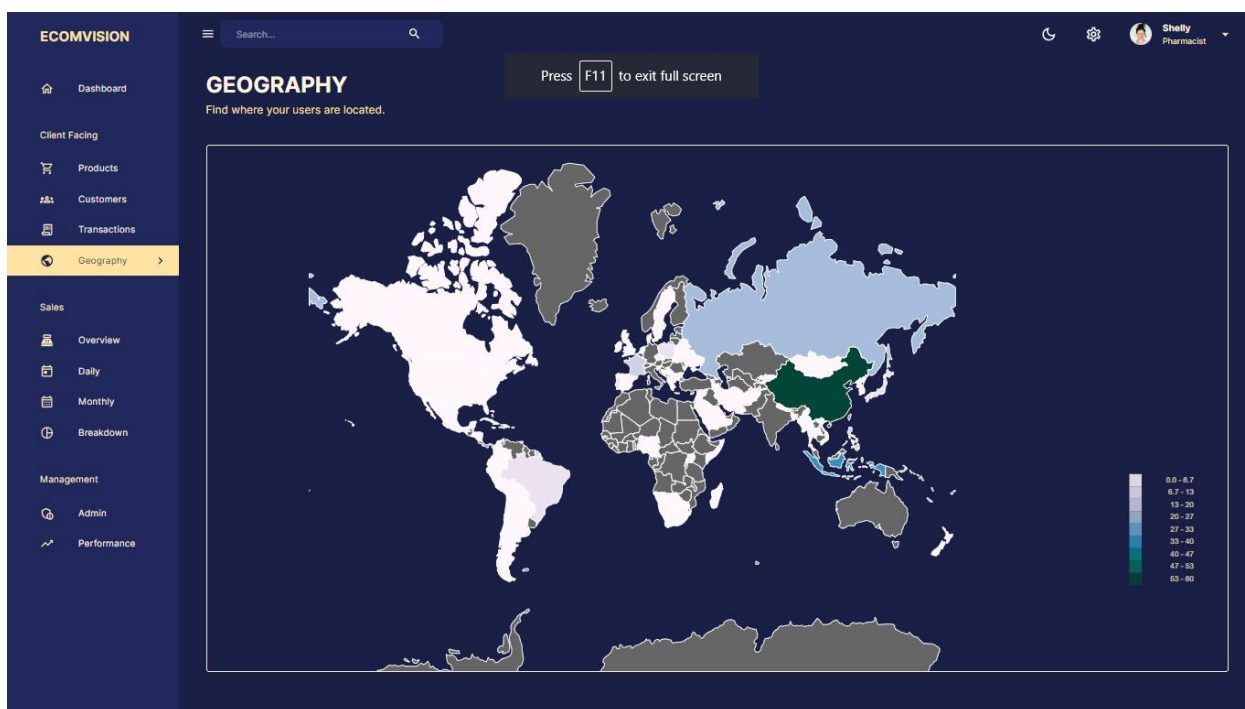


Fig.5. Geography page shows where customers are located

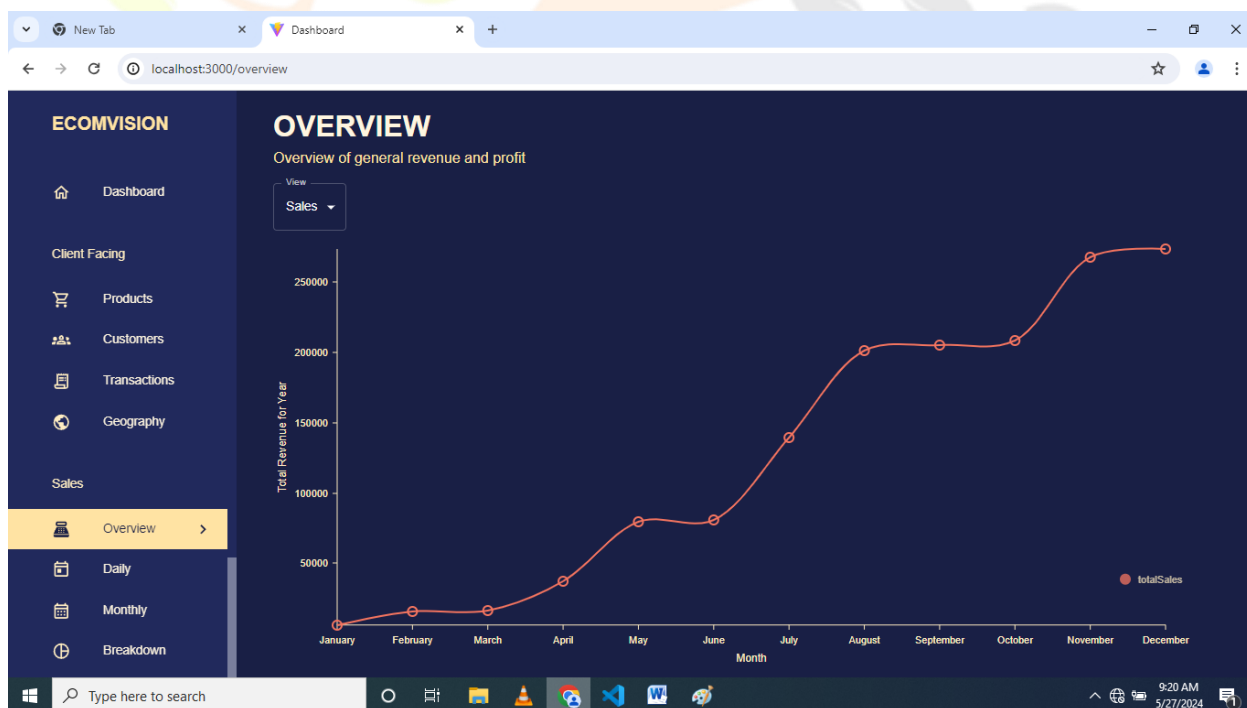


Fig.6. Sales overview by each month

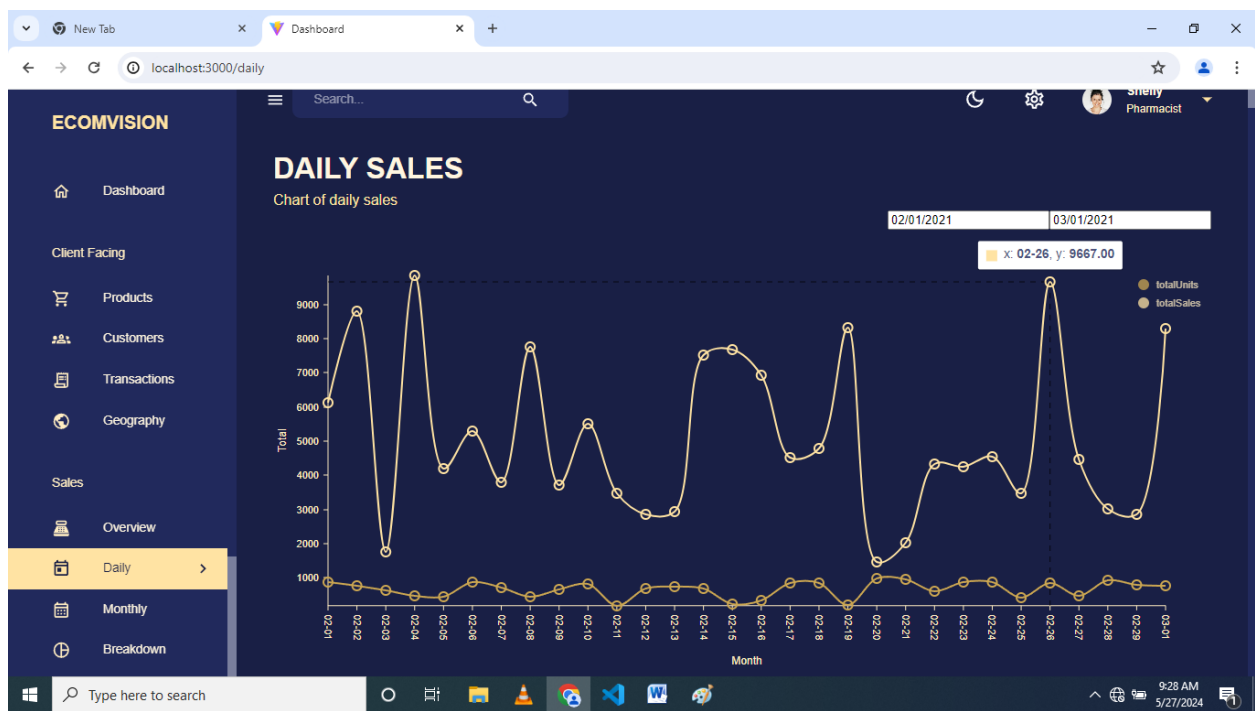


Fig.7. Daily sales analysis page

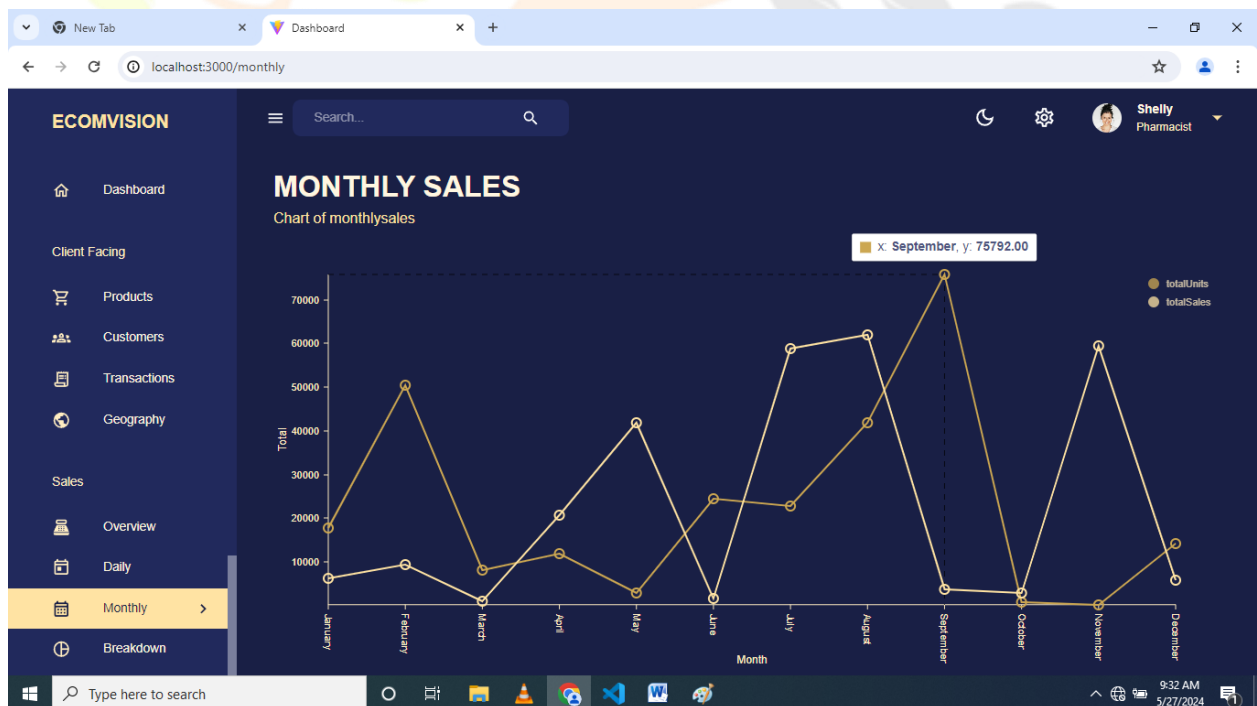


Fig.8 Monthly Sales chart

8. Conclusion:

This application addresses Data visualization problems encountered by businesses using Full Stack MERN with additional frameworks. It seamlessly integrates critical data, employs a variety of data visualization techniques, and analytics to provide valuable insights into Data Visualization. Utilizing modern technologies for both the front-end and backend architecture, MERN Stack Data visualization problems are customizable to produce exact visualization. This application is grounded in extensive research and is designed to empower big data-driven capabilities.

REFERENCES

- [1] A Literature Survey on Data Visualization, Data Extraction, and Data Analysis Nikhil Rathod¹, Saurabh Shinde², Prof. S. R. Kokane
- [2] Information visualization (INFOVIS) // IEEE Symposium. <http://ieeexplore.ieee.org>
- [3] Mashkoor A., Ahamad M.V. Visualization, Security and Privacy Challenges of Big Data // International Journal of Advanced Technology in Engineering and Science, 2017, vol.5, no.6, pp.394–400
- [4] Kumar S. A review of recent trends and issues in visualization // International Journal on Computer Science and Engineering (IJCSE), 2016, vol.8, no.3, pp.41–54.
- [5] Gorodov E., Gubarev V. Analytical Review of Data Visualization Methods in Application to Big Data // Journal of Electrical and Computer Engineering, 2013, pp.1–7.
- [6] Kaushik A., Naithan S. An Anatomy of Data Visualization // International Journal of Computer Science and Network Security, 2016, vol.16, no.2, pp.77–82
- [7] Singh, A., Singh, N., & Gupta, V. (2020). A comprehensive review of MERN stack development for web applications International Journal of Computer Sciences and Engineering, 8(5), 448-454
- [8] FinanceVUE - A MERN STACK FINANCE DASHBOARD APPLICATION, Kaarshnee Dewan, Aadith Lasar, Prof. Bhushan Bhokse, International Journal of Novel Search and Development, volume 9, Issue 4, April 2024.

